
TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2612 – Elektrotechnika a informatika

Studijní obor: 2612R011 – Elektronické informační a řídicí systémy

Automatizace sběru dat z digitálního wattmetru

Automation of data collection from a digital wattmeter

Bakalářská práce

Autor: **Zdeněk Tomšů**

Vedoucí práce: Ing. Lenka Kretschmerová, Ph.D.

Konzultant: doc. Ing. Miroslav Svoboda

V Liberci 18. 5. 2012

ZDE BUDE VLOŽENO ZADÁNÍ

Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum

Podpis

Poděkování

Chtěl bych poděkoval vedoucí mé bakalářské práce Ing. Lence Kretschmerové, Ph.D. za odborné vedení, poskytnuté informace, rady a za čas, který mé práci věnovala. Poděkování patří i konzultantovi doc. Ing. Miroslavu Svobodovi za poskytnuté konzultace. Dále bych rád poděkoval rodičům za umožnění studia. Celé rodině a blízkým přátelům za podporu.

Abstrakt

Bakalářská práce se zabývá automatizací měření výkonů a komunikací po sériové lince. S každou z těchto problematik je podrobněji seznámeno v teoretickém rozboru. Následně je v bakalářské práci uvedeno praktické řešení odečítání dat z digitálního wattmetru. Obsahem řešení je rozbor komunikace RS-232 a vytvoření spojení. Návrh a realizace programu probíhala v grafickém programovacím prostředí LabView. Dále je řešen přístup a ovládání programu ze vzdáleného pracoviště. Součástí práce je navržená laboratorní úloha, která bude plnit účel učební pomůcky. V úloze se ze vzdáleného pracoviště čítají hodnoty wattmetru, kterým je měřena časová stabilita příkonu asynchronního motorku. Pro ověření laboratorní úlohy bylo provedeno vzorové měření.

Klíčová slova: digitální wattmetr, RS-232, LabView, vzdálený přístup

Abstract

This thesis deals with the automation of performance measurement and communication over the serial link. Each of these issues is fully detailed in the theoretical analysis. Subsequently, the thesis provides a practical solution for reading data from a digital power meter. The content of the solution is to analyze the RS-232 and establish a connection. Design and implementation of the program took place in the graphical programming environment called LabView. The access and control from a remote workstation is also an accomplished part of the thesis. The role of the designed laboratory work will serve the purpose of teaching aids. The remote access to the digital power meter is used in the laboratory work for measurement of a time – stability power of an asynchronous motor. To verify the role of the laboratory work sample measurement has been provided.

Key words: digital power meter, RS-232, LabView, remote panel

Obsah

Prohlášení	3
Poděkování	4
Abstrakt	5
Seznam obrázků	10
1 Úvod	13
2 LabView	14
2.1 Historie National Instruments a vývoj LabView	14
2.2 Front Panel	15
2.3 Block Diagram	15
3 Sériová komunikace RS-232C	16
3.1 Střední varianta (9 pinů)	16
4 Rešerše	18
4.1 Elektrický výkon	18
4.1.1 Stejnosměrný proud	18
4.1.2 Střídavý proud	18
4.1.3 Fázorové vyjádření	19
4.1.4 Příkon	19
4.1.5 Účinník	20
4.2 Měření výkonu	20
4.2.1 Měření výkonu stejnosměrného proudu	20
4.2.2 Měření výkonu střídavého proudu	20
4.2.3 Měření trojfázového výkonu souměrné zátěže	21

4.2.4	Měření činného výkonu pomocí dvou wattmetrů.....	22
4.2.5	Měření činného výkonu nesouměrné zátěže	22
4.3	Měřicí přístroje	23
4.4	Laboratorní podmínky vs. praxe.....	24
5	Digitální wattmetr	26
6	Návrh a realizace spojení	27
6.1	Výběr metody připojení.....	27
6.2	Komunikace.....	27
7	Architektura programu	29
7.1	Volitelné módy	29
7.1.1	Simulace	30
7.1.2	RS-232.....	31
7.1.3	Ze souboru.....	33
7.2	Rozšifrování dat.....	34
7.2.1	Desetinná hodnota	36
7.3	Zobrazení na čtyřech displejích.....	39
7.4	Správa programu.....	40
7.4.1	Simulace	40
7.4.2	Ze souboru.....	40
7.4.3	RS-232.....	42
7.5	Sběr dat.....	44
7.5.1	Nastavení sběru dat	44
7.5.2	Tabulka.....	46

7.5.3	Hlavička tabulky	46
7.5.4	Vložení do tabulky	47
7.5.5	Uložení do tabulky	48
8	Přehled pro uživatele	49
8.1	Front panel	49
8.2	Vnitřní struktura	50
9	Vzdálený přístup	51
9.1	Očekávání	51
9.2	Realizace.....	52
9.2.1	Umožnění přístupu z PC-1	52
9.2.2	Zapnutí Web Serveru	54
9.2.3	Navázání spojení z PC-2	55
10	Úprava laboratorní úlohy	57
10.1	Provedení vzorového měření	57
10.2	Naměřené hodnoty	57
11	Shrnutí	59
12	Závěr.....	60
	Použitá literatura	61
	Obsah CD	63
	Příloha A – původní laboratorní úloha	64
	Příloha B – Nová laboratorní úloha.....	66
	Příloha C – Katalogový list wattmetru DW6090	68

Seznam obrázků

Obr. 1: RS-232, 9 pinů	17
Obr. 2: fázorové vyjádření	19
Obr. 3: jednofázová soustava, menší zátěž	20
Obr. 4: jednofázová soustava, větší zátěž	21
Obr. 5: trojfázová soustava, jedním wattmetrem souměrná zátěž s vodičem PE.....	21
Obr. 6: Trojfázová soustava, jedním wattmetrem souměrná zátěž bez PE	22
Obr. 7: trojfázová soustava, Aronovo zapojení.....	22
Obr. 8: trojfázová soustava, třemi wattmetry nesouměrná zátěž s vodičem N	23
Obr. 9: trojfázová soustava, třemi wattmetry nesouměrná zátěž bez vodiče N	23
Obr. 10: Obecné blokové schéma číslicového wattmetru	24
Obr. 11: Lutron DW6090 [POWER ANALYZER, 2000]	26
Obr. 12: Schéma zapojení, Steroe Jack to RS-232.....	27
Obr. 13: Kódování datového slova.....	28
Obr. 14: Volitelné módy, Front Panel	29
Obr. 15: Simulace, Block Diagram	30
Obr. 16: Simulace, Case, Jednotka.....	31
Obr. 17: SubVI Simulace	31
Obr. 18: RS-232, Block Diagram.....	32
Obr. 19: RS-232, Záloha hodnot	32
Obr. 20: RS-232, Sloupec řetězců.....	32
Obr. 21: Ze souboru, Block Diagram	33
Obr. 22: SubVI Ze souboru	33

Obr. 23: SubVI Rozšifrování dat.....	34
Obr. 24: Rozšifrování dat, Block Diagram	35
Obr. 25: Rozšifrování dat, Case, Display	35
Obr. 26: Rozšifrování dat, Case, Polarita.....	36
Obr. 27: Rozšifrování dat, Case, Jednotka	36
Obr. 28: SubVI Desetinná hodnota	36
Obr. 29: Desetinná hodnota, Block Diagram	37
Obr. 30: Desetinná hodnota, Block Diagram, náhled na výpočet	37
Obr. 31: Desetinná hodnota, Case, desetinná čárka	38
Obr. 32: Zobrazení na čtyřech displejích, Front Panel.....	39
Obr. 33: Zobrazení na čtyřech displejích, Case, Display 1-4.....	39
Obr. 34: Simulace, Správa.....	40
Obr. 35: Simulace, Block Diagram	41
Obr. 36: Ze souboru, Správa	41
Obr. 37: Ze souboru, Block Diagram	42
Obr. 38: Ze souboru, Case.....	42
Obr. 39: RS-232, Správa	42
Obr. 40: RS-232, Block Diagram.....	43
Obr. 41: RS-232, Case.....	43
Obr. 42: Sběr dat, aktualizace hodnot + Case	44
Obr. 43: Nastavení sběru dat, Front Panel	45
Obr. 44: Nastavení sběru dat, zakázání uložení	45
Obr. 45: Tabulka, Front Panel.....	46

Obr. 46: Hlavička tabulky, Front Panel.....	46
Obr. 47: Hlavička tabulky, Block Diagram.....	47
Obr. 48: Vložení do tabulky, Block Diagram	48
Obr. 49: Vložení do tabulky, Case, jednotlivá pole	48
Obr. 50: Uložení do tabulky, Front Panel	48
Obr. 51: Front Panel	49
Obr. 52: Block Diagram	50
Obr. 53: Enable Remote Panel Server.....	52
Obr. 54: Visible VI's.....	53
Obr. 55: Browser Access.....	53
Obr. 56: Web Server VI name.....	54
Obr. 57: Web Server titulky	54
Obr. 58: Web Server Save to Disk	55
Obr. 59: Web Server OK.....	55
Obr. 60: Connect to Remote Panel.....	56
Obr. 61: Tabulka hodnot	57
Obr. 62: Časová stabilita příkonu.....	58
Obr. 63: Časová stabilita účinníku	58

1 Úvod

Automatizace se postupně prosakuje do všech odvětví a jednou bude součástí i každé evropské domácnosti. Využíváme ji pro ulehčení práce. Ve spoustě činností nahradily stroje lidskou sílu. Další možností je úspora energie, ovládané světelné okruhy, systematické zapínání a vypínání spotřebičů. Dalším z mnoha aspektů je i zajištění bezpečí, jak už v továrních prostorách tak v bytech a rodinných domech. Nejčastěji však automatizace zabezpečuje více aspektů zároveň.

Praktická činnost je se dá rozdělit na několik částí. V první řadě je mým úkolem seznámit se s digitálním wattmetrem Lutron DW6090 a připojit ho k PC. Další částí je zabezpečení komunikace. Wattmetr komunikuje přes sériovou linku RS-232C. Následuje sběr dat, která wattmetr posílá. Tyto data je třeba rozšifrovat a zobrazovat v uživatelsky přívětivém prostředí. Pro vytvoření programu je v práci využito grafické programovací prostředí LabView. Pomocí stejného programu je zajištěna možnost vzdáleně ovládat pracoviště. Tyto úkony jsou posléze využity při návrhu laboratorní úlohy pro předmět MT1.

Práce návrhářská spočívala v navržení struktury komunikace. Mechanická činnost je zastoupena vytvořením kabelu pro komunikaci. Práce programátorská spočívala v navržení struktury programu, naprogramování a nastavení vzdáleného přístupu. Další činností byla úprava laboratorní úlohy a provedení jejího vzorového měření.

Veškeré problematiky, které se praktické části týkají, nebo se kterými se pracovalo, jsou nastíněny z kraje bakalářské práce. Je to programovací prostředí, komunikace po sériové lince a měření výkonu.

2 LabView

Je grafické vývojové prostředí pro tvoření aplikací. Uživatel dokáže i bez větších zkušeností rychle a efektivně používat měřicí a řídicí hardware, analyzovat data, sdílet výsledky a distribuovat systémy.

Prostředí obsahuje dvě základní části, které jsou spolu úzce svázány. Jsou to Front Panel a Block diagram.

2.1 Historie National Instruments a vývoj LabView

V roce 1976 byla založena firma s názvem National Instruments v Texasu. Tuto společnost založil James Truchard, Jeffrey Kodovsky a William Nowlin, právě když společně pracovali na sonaru pro americké námořnictvo. Při řešení zakázky vznikla myšlenka „virtuální instrumentace“. Firma se během čtyř let rozrostla do takové míry, že všichni zakladatelé se stali zaměstnanci na plný úvazek. V roce 1986 Jeffrey Kodovsky začíná vyvíjet program LabView. V následujícím roce je zaměstnaný jubilejní stý zaměstnanec a začátkem následujícího roku je na trhu uvedena vůbec první verze programu LabView s názvem LabWindows Version 1.0 for DOS. Paralelně také vyrábí plug-in zařízení pro Apple Macintosh a IBM. S rokem 1990 přichází další dvě verze LabView a to LabWindows Version 2.0 a LabView Version 2.0. Během příštích 7 let vydají ještě několik verzí a jejich produkty se v automatizační technice prosazují čím dál tím více. Do roku 2005 se z jedné kanceláře o rozloze 28 m² rozšířili na 3 budovy o celkové rozloze kolem 70 000 m² a výrobní závod v Maďarsku. Právě rokem 2005 vychází už 8 verze programu LabView. Během příštích 2 let oslaví firma 30. a program LabView 20. výročí, vyjde LabView verze 8.5 a pomocí Measurement Studio 8 je možné monitorovat a řídit své aplikace vzdáleně přes internet. V dnešní době se firma zase o něco rozšířila. Nabízí širokou škálu softwarových a hardwarových produktů. V dnešní době už existuje jedenáctá verze programu NI LabView 2011. Jedním z největších úspěchů je využití LabView vědci z NASA pro analýzu a zobrazení dat teploty, polohy, stavu baterií a monitorování celkového stavu vozítka Mars Pathfinder Sojourner, jezdící právě na povrchu Marsu. Jiné aplikace například řídí a monitorují chod elektráren, testují mobilní telefony či je jich využito k řízení urychlovače částic ve švýcarském CERN.

Jeffrey Kodovsky řídil vývoj LabView podle myšlenky, nikdy neděláme pouze jednu věc. Svět sám o sobě je tedy paralelní. Když na svět začaly prorážet více-jádrové procesory,

byl program LabView připraven. Tzv.: více-vláknová architektura perfektně pasuje právě na více-jádrové procesory. Paralelní programování spolu s více-jádrovými procesory vše výrazně urychlilo. V institutu Maxe Plancka pro fyziku plazmatu bylo dosaženo dvacetinásobného zrychlení s osmi-jádrovým procesorem oproti procesoru s jedním jádrem.

2.2 Front Panel

Okno slouží hlavně jako ovládací a informační pult. Nejčastěji obsahuje různá grafická tlačítka, vypínače *Control*, ukazatele v nejrůznějších formách *Indicator* nebo například grafy a pole.

2.3 Block Diagram

Tato část obsahuje minimálně stejné prvky jako Front Panel pouze v jiné formě. A my tak můžeme tyto bloky propojovat, upravovat jejich defaultní nastavení. Další možnosti tohoto prostředí jsou programovací smyčky, jako jsou například For, While a If, předdefinované komunikace s hardwarem nebo universální komunikace pomocí VISA.

Při tvoření a ladění programu se pracuje hlavně v Block Diagram. Při spuštění programu se využívá pouze Front Panel.

3 Sériová komunikace RS-232C

Americká asociace pro elektronický průmysl EIA v roce 1962 normovala toto rozhraní pod názvem EIA RS-232. Později prodělalo jen doplňkové úpravy a získalo příponu „C“, celým názvem EIA RS-232C. V praxi se používá pouze RS-232 nebo možná pro někoho méně známé rozhraní V.24. Je jedním z nejstarších rozhraní, které umožňovalo sériový asynchronní přenos dat. Prvotním důvodem vzniku bylo připojení modemu k počítači. Byly vytvořeny 3 typy.

- **Malá varianta**

Umožňovala jednakanálový asynchronní přenos dat bez zjištění stavu zařízení.

- **Střední varianta**

Je opatřena 9 piny a umožňuje jak synchronní tak i asynchronní přenos. Hodí se pro krátkou vzdálenost (do 15m), pro delší vzdálenost při použití modemu. Jako jediná se až na malé výjimky používá do dnes.

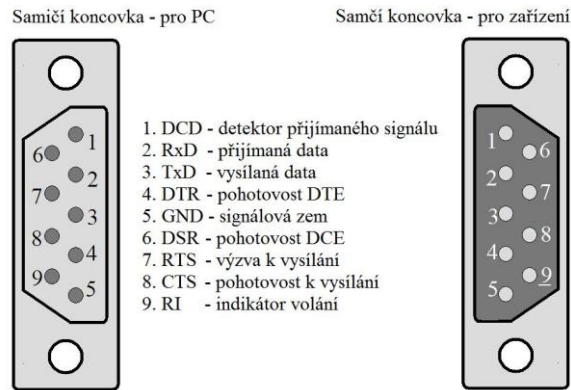
- **Velká varianta**

Zahrnuje 25 pinů a má podobné schopnosti jako střední varianta.

Následně se takto připojovaly i jiné periferie jako např.: myš, mobilní telefon či tiskárna. Dnes je pro tyto zařízení rozhraní zastaralé a stále více se používá USB (Universal Serial Bus), tedy univerzální sériová sběrnice. Přesto sériová linka stále nachází uplatnění u nestandardních zařízení. Díky své konstrukci umožňuje rychlý a bezkonfliktní přenos dat. V neposlední řadě také nabízí jednoduché ovládání z hlediska programování komunikace mezi PC a připojeným zařízením. Je tak v hodné i pro jednoduché pokusy.

3.1 Střední varianta (9 pinů)

Přenos dat je zastoupen piny TxD a RxD. Ostatní piny plní funkce strukturování a řízení přenosu dat. Používají se pro vzájemné potvrzování platnosti přenášených dat. Díky tomu je možné programově číst nebo nastavovat jejich stavy. Další výhodou je, že ze sériového rozhraní můžeme odebírat napětí a rovnou tak napájet menší zařízení. Podrobnější vysvětlení pinů je vidět na Obr. 1.



Obr. 1: RS-232, 9 pinů

Vstupy konektoru mají výstupní odpor přibližně $10k\Omega$. Rozeznávají napětí pod hodnotu $-3V$ jako log. 1 a napětí vyšší než $+3V$ jako log. 0. Je tak zabezpečeno, že na přenos dat nebude mít nežádoucí účinky hystereze, která se dostavuje při přepínání stavů. Ovládání zajišťují binární úrovně, které jsou zastoupeny $-12V$ pro log. 1 a $+12V$ pro log. 0. V každém sériovém portu PC se používá několik registrů s adresovatelnými bity, které zajišťují říditelnost veškerých funkcí sériového vstupu či výstupu.

Výhody RS-232C

- odolnost proti mechanickému zničení
- možno připojovat zařízení i při zapnutém PC
- napájecí napětí přímo z portu
- programově jednoduše ovládatelné

4 Rešerše

Rešerše pojednává o měření výkonu v laboratorních podmínkách a v praxi. Začíná krátkým seznámením s elektrickým výkonem a veličinami s ním spojenými. Dalším bodem jsou možnosti měření výkonu a téma zaobírající se rozdílností laboratorních podmínek a praxe.

4.1 Elektrický výkon

Je fyzikální veličinou, která udává, kolik vykoná elektrická síla práce za jednotku času. Jednotkou výkonu je watt. Pojmenovaný podle skotského inženýra Jamese Watta a je označován, jako velké písmeno W. Hodnota jednoho wattu odpovídá stejnoměrnému proudu 1A při úbytku napětí 1V. Elektrický výkon dělíme do několika skupin: činný, jalový a zdánlivý, popřípadě deformační a komplexní. Samozřejmě se liší výkon stejnosměrného a střídavého proudu.

4.1.1 Stejnosměrný proud

V případě stejnosměrného proudu je proud i napětí neměnné. Je tedy výkon v každém čase stejný. Lze tak výkon vyjádřit vztahem (1), popřípadě s využitím Ohmova zákona rovnicí (2).

$$P = p(t) = UI; P = \frac{U^2}{R} = I^2 R \quad (1) \text{ a } (2)$$

4.1.2 Střídavý proud

U střídavých proudů jsou veličiny závislé na čase. Jejich průběhy se nejčastěji dají vyjádřit jako periodické sinusové průběhy. Pomocí vyjádření proudu a napětí v sinusovém tvaru s jejich amplitudou, úhlovou frekvencí a fázovým posuvem, platí vyjádření rovnice (3).

$$\begin{aligned} p(t) &= u(t)i(t) = U_m \sin(\omega t + \varphi) I_m \sin(\omega t) = \\ &= UI((1 - \cos 2\omega t)\cos\varphi + \sin 2\omega t \sin\varphi) \end{aligned} \quad (3)$$

Podle jednoduchého vzorce se můžeme dopočítat efektivních hodnot. Uvedeno v rovnicích (4) a (5).

$$\sqrt{2}U = U_m; \sqrt{2}I = I_m \quad (4) \text{ a } (5)$$

Činný výkon

$$P = UI \cos\varphi [W]; I_c = I \cos\varphi \quad (6) \text{ a } (7)$$

Jalový výkon

$$Q = UI \sin\varphi [Var]; I_j = I \sin\varphi \quad (8) \text{ a } (9)$$

Zdánlivý výkon

$$S = UI [VA] \quad (10)$$

4.1.3 Fázorové vyjádření

komplexní výkon je definovaný vztahem (11)

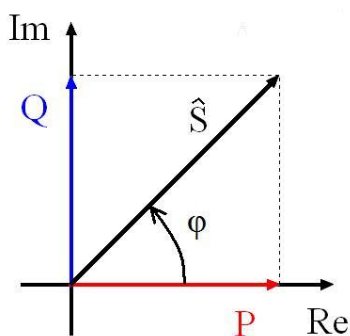
$$S = UI^* = P + jQ \quad (11)$$

Následně lze vyjádřit na základně poznatků z rovnice (11)

$$P = \operatorname{Re}(S) = \frac{1}{2} \operatorname{Re}(U_m I_m^*) \quad (12)$$

$$Q = \operatorname{Im}(S) = \frac{1}{2} \operatorname{Im}(U_m I_m^*) \quad (13)$$

$$S = |S| = \frac{1}{2} |U_m I_m^*| \quad (14)$$



Obr. 2: fázorové vyjádření

4.1.4 Příkon

Výkon přístroje zahrnuje práci, která je potřeba čistě pro požadovaný účel, vše ostatní (například: výkon, který se spotřeboval na ohřátí žárovky) zahrnuje právě příkon. Udává energetickou náročnost, jednotkou je Watt.

4.1.5 Účinník

Značkou účinníku je $\cos \varphi$, je bezrozměrnou veličinou. Nabývá hodnot z intervalu $<0;1>$ častěji potom díky ztrátám z intervalu $(0;1)$. Vyskytuje se pouze u střídavého proudu a pomocí něj se vypočítává činný výkon. Účinník vyjadřuje podíl činného a zdánlivého výkonu. Určuje kolik užitečné energie je možno získat ze zdánlivého výkonu. Je přímo závislý na fázovém rozdílu proudu a napětí. Ideální hodnota účinníku je 1. Čím je číslo menší, tím větší jsou v obvodu ztráty energie. Běžně by se $\cos \varphi$ měl pohybovat kolem hodnot 0,95 až 0,99.

4.2 Měření výkonu

Měření se rozděluje druhem proudu, zda se jedná o stejnosměrný nebo střídavý proud. Pro střídavý proud se měření dále člení podle počtu fází.

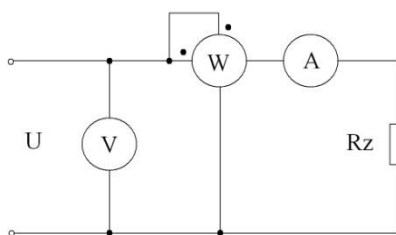
4.2.1 Měření výkonu stejnosměrného proudu

Při měření výkonu stejnosměrného proudu voltmetrem a ampérmetrem se dopouštíme chyby způsobené vlastní spotřebou měřících přístrojů. Při měření velkých výkonů lze tuto chybu zanedbat, jinak je třeba provést opravu. Oprava se počítá podle zapojení měřících přístrojů a podle toho, měříme-li příkon spotřebiče nebo výkon zdroje. Jinými slovy je potřeba odečíst příkon voltmetru a ampérmetru.

4.2.2 Měření výkonu střídavého proudu

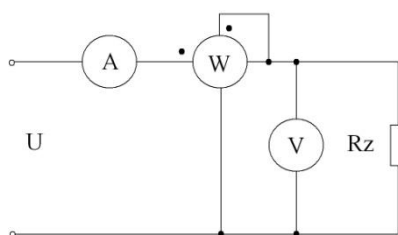
Činný výkon můžeme měřit pomocí přístroje známého jako wattmetr, který může být analogový nebo digitální. Jalový výkon lze měřit pouze upraveným wattmetrem, který má fázově posunuté připojené napětí o $\frac{1}{2}\pi$.

Jednofázová soustava:



Obr. 3: jednofázová soustava, menší zátěž

Na Obr. 3 je vstupní napěťová svorka spojená se vstupní proudovou svorkou wattmetru. Wattmetr měří i výkon své proudové cívky. Oprava se neprovádí, protože obvykle není znám odpor proudové cívky.



Obr. 4: jednofázová soustava, větší zátěž

Na Obr. 4 je vstupní napěťová svorka je spojena s výstupní proudovou cívkou a wattmetr měří i výkon v napěťové cívce. Při měření malých výkonů provedeme opravu vztahy (15) a (16).

$$P_S = P_N - \frac{U^2}{R_N} \quad (15)$$

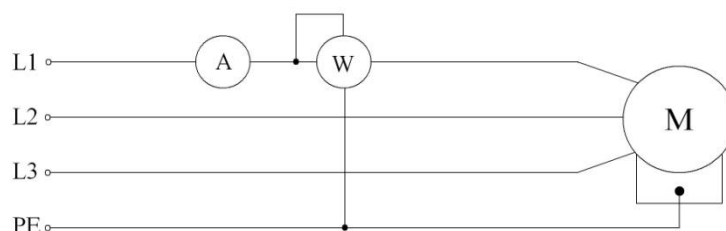
$$R_N = \frac{R_W R_V}{R_W + R_V} \quad (16)$$

R_V – odpory napěťových cívek měřících přístrojů

Stále se ale jedná o soustavu pouze s jednou fází. V případě více-fázové soustavy (tzv. n-fázové) je potřeba $n - 1$ wattmetrů, v případě trojfázové soustavy je pak potřeba minimálně dvou wattmetrů. Jedná se o tzv. Blondelův teorém.

4.2.3 Měření trojfázového výkonu souměrné zátěže

V případě souměrného zatížení fází je možné měřit výkon pouze jedním wattmetrem na Obr. 5. Ve výsledku pak výkon vynásobit třemi jak je uvedeno v rovnici (17).

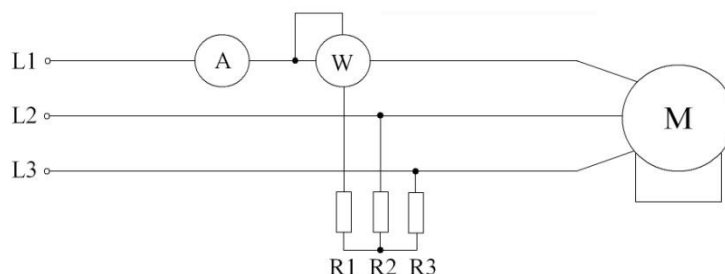


Obr. 5: trojfázová soustava, jedním wattmetrem souměrná zátěž s vodičem PE

$$P = 3 \text{ kw } \alpha \quad (17)$$

kw – konstanta wattmetru

V případě souměrné zátěže a nevyvedeného středního vodiče na Obr. 6 je možné vytvoření umělé nuly pomocí trojice odporů, které musí odpovídat rovnici (18). Pro vypočtení výkonu se využije rovnice (19).



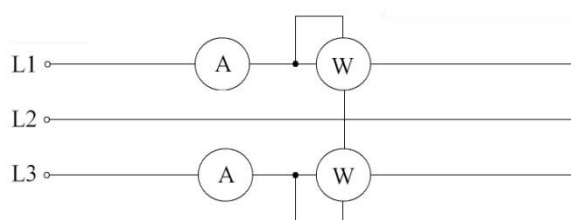
Obr. 6: Trojfázová soustava, jedním wattmetrem souměrná zátěž bez PE

$$R_1 + R_W = R_2 = R_3 \quad (18)$$

$$P = 3 kw \alpha \quad (19)$$

4.2.4 Měření činného výkonu pomocí dvou wattmetrů

Pomocí schématu na Obr. 7 (tzv. Aronovo zapojení) je možné pomocí dvou wattmetrů měřit jak činný tak jalový výkon. Napěťové cívky wattmetrů musí být připojeny na sdružené napětí. Matematický popis je uveden v rovnicích (20) a (21).



Obr. 7: trojfázová soustava, Aronovo zapojení

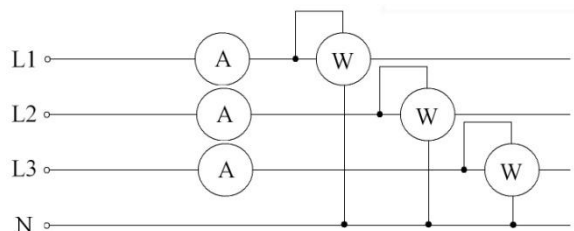
$$P = kw(\alpha_1 + \alpha_2) \quad (20)$$

$$Q = kw(\alpha_1 - \alpha_2) * \sqrt{3} \quad (21)$$

4.2.5 Měření činného výkonu nesouměrné zátěže

Nesouměrná zátěž se měří pomocí třech wattmetrů. Měřením jedním wattmetrem dochází k nepřesnostem daným různou velikostí výkonu jednotlivých fází, a tak se tento způsob nedoporučuje.

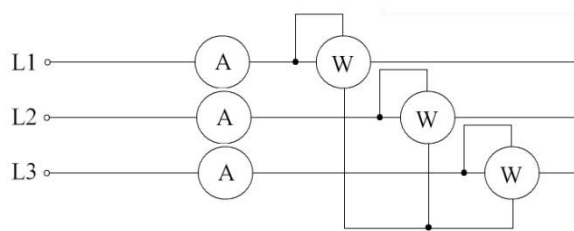
V případě nesouměrné zátěže je využito třech wattmetrů připojených na společnou zem (viz Obr. 8). Výsledek rovnice (22) nám dává celkový výkon.



Obr. 8: trojfázová soustava, třemi wattmetry nesouměrná zátěž s vodičem N

$$P = kw(\alpha_1 + \alpha_2 + \alpha_3) \quad (22)$$

V případě nevyvedeného středního vodiče se používá vytvoření umělé nuly spojením samotných wattmetrů, jak je ukázáno na Obr. 9. Výpočet celkového výkonu se provádí podle rovnice (23).



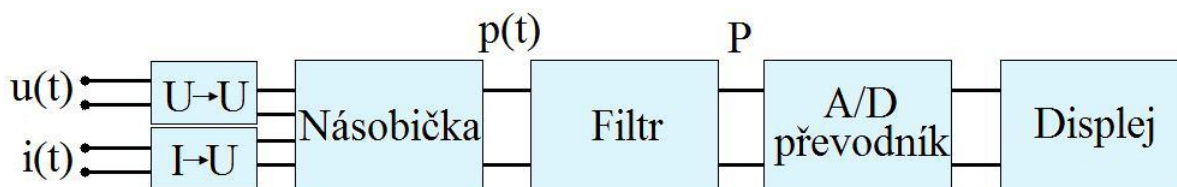
Obr. 9: trojfázová soustava, třemi wattmetry nesouměrná zátěž bez vodiče N

$$P = kw(\alpha_1 + \alpha_2 + \alpha_3) \quad (23)$$

4.3 Měřicí přístroje

Pro měření el. výkonu se používá jak digitálních tak i analogových přístrojů. U analogových zvláště elektrodynamické a ferodynamické měřicí přístroje.

- **Elektrodynamické** – pevná proudová cívka (v sérii se zátěží) ovlivňuje napětovou otočnou cívku (zapojena paralelně k zátěži), která způsobuje výchylku ručky. Dosahují třídy přesnosti 0,1, 0,2 i 0,5.
- **Ferodynamické** – stejný princip jako u elektrodynamického přístroje, s tím rozdílem že obsahuje feromagnetické jádro. Otočná cívka je tak ve vzduchové mezeře tohoto jádra. Třída přesnosti 0,5 a 1.
- **Číslicové** – obsahují proudový a napětový vstup. Při vyšších frekvencích dosahují lepší přesnosti než analogové přístroje, které jsou značně frekvenčně závislé. Princip činnosti je naznačen na Obr. 10.



Obr. 10: Obecné blokové schéma číslicového wattmetru

4.4 Laboratorní podmínky vs. praxe

Měření výkonu v laboratoři slouží pro výuku a technologický rozvoj v rámci vývoje nových technologií. Naopak v praxi se spíše než výkon měří elektrická práce daná vztahem (24).

$$W = \int_0^T P * dt \text{ [kWh]} \quad (24)$$

K měření elektrické práce dochází z kontrolních důvodů, ale také pro vyúčtování spotřebované el. energie. Pro toto měření se využívá watthodinových elektroměrů (tzv. elektrických hodin). Princip funkce je podobný jako u analogových wattmetrů. Obsahují proudovou a napěťovou cívku. Avšak jejich otočné ústrojí nemá direktivní moment. Otáčení pracuje na rovnováze pohybového a brzdicího momentu, který se využívá k otáčení hliníkového kotoučku. Kotouček je napojen na bubínkový počítací strojek a tak je výstupem z elektroměru číselná hodnota. Elektroměry lze rozdělit do tří skupin.

Rozdělení podle druhu proudu

- pro stejnosměrný proud
- pro střídavý proud

Elektroměry pro střídavý proud lze dělit

- jednofázové
- trojfázové

Rozdělení podle druhu měřené práce

- elektroměry pro měření činné práce
- elektroměry pro měření jalové práce
- elektroměry pro měření zdánlivé práce

Dnes jsou k dostání i digitální přístroje zapojené jako meziblok mezi spotřebičem a zásuvkou. Tyto přístroje měří okamžitou spotřebu přístroje, který je v nich zapojený, tedy výkon. Uživatelé poskytují spíše orientační hodnoty.

Jak v domácnosti, tak i v průmyslové oblasti je veliký zájem o přehled spotřebované elektrické energie. V obou případech se může jednat o největší finanční výdaje a je logické, že cílem spotřebitelů je tyto výdaje snížit na minimum. Dalším důležitým důvodem, proč snížit spotřebu el. energie, by mělo být šetrné chování k životnímu prostředí. Pokud se nejedná o získávání el. energie z obnovitelných zdrojů.

V průmyslové praxi se tak využívá např.: inteligentních řídicích systémů pro regulaci osvětlení v halových prostorech, automatizace skladů, využívání energeticky méně náročných strojů. V domácnosti je veliký důraz kladen na spotřebiče z vyšších energetických tříd. Tyto přístroje jsou oproti jiným dražší, avšak návratnost investice se vyplácí během několika prvních let. Tyto třídy jsou označovány písmeny A až G, kdy A je nejúspornější energetická třída. Naposledy do této skupiny přibýly i třídy A+ a A++.

5 Digitální wattmetr

Stolní multifunkční měřicí přístroj (viz Obr. 11) je opatřen digitálním LCD displejem o rozměrech 93 x 52 mm. Je schopen měřit W, VA, Wh, $\cos \varphi$, AC V, AC A, DC V, DC A, Hz a Ω . Umožňuje současné zobrazení výkonu, účinníku, napětí a proudu. Automaticky volí rozsahy a pro připojení k PC je opatřen výstupním rozhraním RS-232.



Obr. 11: Lutron DW6090 [POWER ANALYZER, 2000]

Technické údaje:

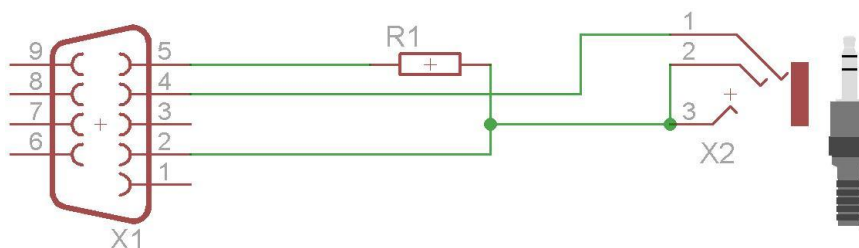
Veličina	rozsah	přesnost
W přímý vstup	6kW	$\pm(1,5\% + 5d)$
W indukt. sonda	9,9kW ÷ 999,9kW	$\pm(1,5\% + 5d)$
VA přímý vstup	99,9kW ÷ 999,9kW	$\pm(2\% + 2d)$
Účinník	0,01 ÷ 1,00	$\pm(1,5\% + 2d)$
AC \ DC V	0,1V ÷ 600V	$\pm(1,0\% + 1d)$
AC \ DC A přímý vstup	0,01A ÷ 10,0A	$\pm(1,0\% + 1d)$
AC \ DC A indukt. sonda	1000A	přístroj + sonda
AC A CT transformátor	0,1 ÷ 200A (CT 100/5A)	přístroj + CT
	1 ÷ 2000A (CT 1000/5A)	přístroj + CT
Wh	0,001Wh ÷ 1000kWh	$\pm(1,5\% + 5d)$
Odpor	9,999 Ω ÷ 19,99 k Ω	$\pm(1,0\% + 1d)$
Frekvence	10Hz ÷ 999 Hz	$\pm(1,0\% + 1d)$
Napájení	DC 9V, 1,5 V AA baterie 6 ks	

6 Návrh a realizace spojení

Z hlediska návrhu realizace spojení, plyne ze strany wattmetru jasné omezení. Je schopný komunikovat pouze jediným jednosměrným rozhraním.

6.1 Výběr metody připojení

Digitální wattmetr je od výrobce opatřen jedním komunikačním výstupem. Jedná se o konektor 3,5 JACK. Komunikační spojení je tedy doporučeno kabelem, který má na prvním konci devíti pinovou koncovku DB9, též známou pod označením CANON 9. Opačný konec je opatřen 3,5mm MONO JACK konektorem. Tento kabel je uváděn pod označením UPCB-02. Z hlediska ceny originálního kusu, bylo nasnadě vyrobit si kabel vlastní. Pro jednoduchost bylo použito 3,5mm STEREO JACK konektoru a správným kabelovým propojením dosaženo stejného chování jako v případě 3,5mm MONO JACK. Vrchol konektoru (kontakt 1) 3,5mm STEREO JACK byl připojen na 4. pin (DTR) konektoru CANON 9 v provedení samice. Střední a spodní vodivá ploška (kontakt 2 a 3) jsou spojeny na 2. pin CANON 9 (RxD – pin pro čtení příchozí informace) a přes odpor $R1 = 2,2k\Omega$ na 5. pin (signálová zem). Schéma zapojení na Obr. 12.



Obr. 12: Schéma zapojení, Steroe Jack to RS-232

Pro zabezpečení komunikace bylo vybráno grafické programovací prostředí LabView. Tento software je vhodný pro komunikaci a ovládání externích zařízení. Sám osobně disponuje řadou programovacích nástrojů, které jsou přímo určený pro komunikaci po sériové lince.

6.2 Komunikace

Wattmetr disponuje pouze jednosměrnou komunikací. Odesílá datové balíčky o velikosti 16 bytů. Každý byte nebo kombinace bytů nese informaci o velikosti hodnoty, polaritě nebo pozici, kde je zobrazena.

1. První byte je veden pod názvem startovací slovo.
2. Druhý byte nese vždy stejnou informaci pro správnou orientaci celého balíčku. Tato informace je číslo 4.
3. Třetí byte s hodnotou 1-4 určuje, na který displej se má informace zobrazit.
4. Byte 4. a 5. určuje, v jakých jednotkách byla hodnota naměřena.
6. Šestý byte ne pouze hodnotou 1 nebo 0, určuje polaritu změřené hodnoty. Znaménko mínus (resp. plus) je zastoupeno číslem 1 (resp. 0).
7. Sedmý byte nese jednomístnou hodnotu. Toto číslo určuje, před kolikátým číslem zprava doleva změřené hodnoty, se má nacházet desetinná čárka. Toto číslo se pohybuje v rozsahu <0;3>.
8. Osmý až patnáctý byte nese číselnou informaci o velikosti naměřené veličiny
16. Šestnáctý byte zastupuje konec slova

Program dále pracuje pouze se šestnácti místným textovým řetězcem, kde první a poslední byte je vypuštěn, a všechny ostatní hodnoty zůstávají stejné. Příkladem takové informace je Obr. 13. Křížky ve tvaru písmene X na začátku (resp. na konci) zastupuje *start bit* (resp. *stop bit*). Na obrázku se jedná o kladnou hodnotu výkonu, která má být zobrazena na druhém displeji, hodnotou je +3,41W.



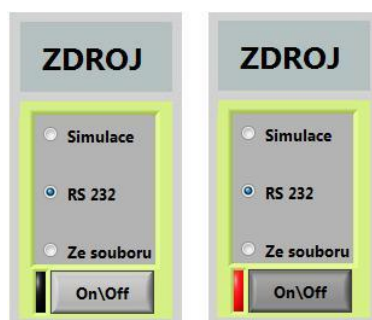
Obr. 13: Kódování datového slova

7 Architektura programu

Program je hned na začátku větven na 3 uživatelsky volitelné módy. Avšak výstupem z těchto módů je vždy datové slovo o stejné struktuře. Liší je akorát hodnotou, kterou nabývají. Následně rozšifrováním tohoto slova, jsou vytvořeny parametry. Pomocí nich je složena konečná podoba změřené veličiny, která se bude zobrazovat. To probíhá na čtyřech displejích, o čemž rozhoduje jeden z právě zmiňovaných parametrů. Jsou tedy zobrazovány maximálně čtyři veličiny zároveň. Po uložení aktuálních hodnot do externího souboru, se vždy aktualizuje tabulka zobrazující náhled na externí soubor.

7.1 Volitelné módy

V části *ZDROJ* se nachází pole *Radio Buttons*, kde jsou vloženy tři zaškrťovací volby. Uživatel si může vždy zvolit pouze jedinou možnost. Svou volbu potvrdí tlačítkem *On\Off*. Rozsvícení červené kontrolky indikuje spuštění programu. Vypnutá i zapnutá varianta je vyobrazena na Obr. 14.



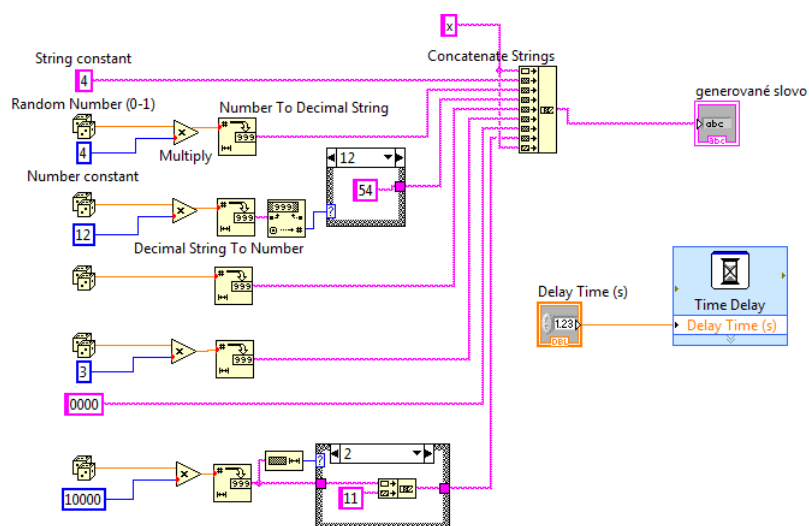
Obr. 14: Volitelné módy, Front Panel

Prvním módem je simulace, která generuje náhodná čísla a je uváděna pod stejným názvem *Simulace*. Dále je volba, pojmenovaná jako *RS-232*, která uskutečňuje zadanou problematiku. Posledním je mód, realizující zpětné projití uskutečněného měření, pod názvem *Ze souboru*. Pro první a třetí mód byly vytvořeny vlastní bloky, definované ikonou a vlastními terminály (tj. vstupy a výstupy). Díky tomu jsou dále v programu používány tzv. SubVI. Přístup k nim je přes zmiňované terminály.

7.1.1 Simulace

Tato část náhodně generuje šestnácti místné řetězce, které napodobují informaci odeslanou wattmetrem v datovém balíčku. Tato čísla jsou v rozsahu zobrazovacích schopností wattmetru. Nedochází tedy ke kolizi a vždy je zobrazen pouze znak, který by mohl nastat i při skutečném měření. Obr. 15 ukazuje, jak vypadá struktura zapojení.

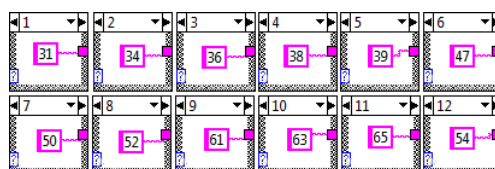
Náhodnost v určeného rozmezí je zajištěna generováním náhodného čísla a následným vynásobením správnou hodnotou. Výstup z bloku *Random Number*, který generuje číslo v rozmezí $<0;1>$ na 2 desetinná místa, je přiveden společně s číselnou konstantou do bloku *Multiply*. Zde se tyto dvě hodnoty spolu vynásobí a tato informace pokračuje do konvertoru *Number to Decimal String*. Tento blok mění číselnou hodnotu z formátu *double* na text ve formátu *string*. Tímto konvertováním nedochází pouze ke sjednocení formátů informací, ale také k zaokrouhlení hodnoty.



Obr. 15: Simulace, Block Diagram

Obdobně se tak získávají všechny hodnoty kromě následujících pěti.

- Informace zastupující druhý byte datového balíčku je nahrazena konstantním symbolem čísla 4 pomocí bloku *String constant*.
- Hodnota určující jednotku měřené veličiny (3. a 4. číslo) z důvodu nesouvislého rozmezí hodnot nelze generovat. Generuje se tedy hodnota 1-12. Touto hodnotou je inicializována smyčka *Case*, která pro každou hodnotu má určené jiné číslo. Bližší vysvětlení je na Obr. 16, který vyobrazuje všechny možné stavy smyčky *Case*.



Obr. 16: Simulace, Case, Jednotka

- Změřená hodnota (7. až 15. číslo) je rozdělena na dvě stejně velké části. První je nahrazena konstantou „0000“. V druhé části je generováno číslo, jako u zbylých hodnot. Avšak cílem je získat čtyř cifernou hodnotu. V případě špatného rozměru čísla, smyčka *Case* doplní číslo řadou jedniček, tak aby bylo číslo čtyř místné. Posléze se tyto 2 části spojí do informace o velikosti 8 znaků.
- První a poslední znak je tvořen znakem *x*.

Následně se všechny znaky spojí blokem *Concatenate Strings* do jedné šestnáctimístné informace.

Pro ovladatelnost rychlosti generování těchto hodnot je zde uveden blok, *Time Delay* ovladatelný blokem *Delay Time*. Hodnotou řídicí bloku je možné určit, k jakému zpoždění v sekundách dojde.

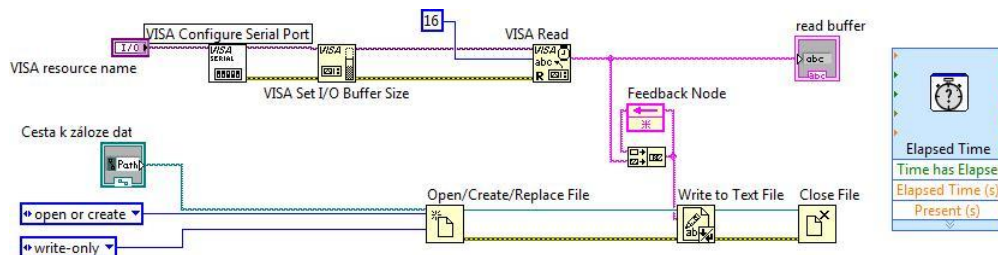


Obr. 17: SubVI Simulace

Na Obr. 17 je vyobrazen vzhled bloku spolu s jeho terminály. Vstupem je *Delay Time* a výstupem *generované slovo*.

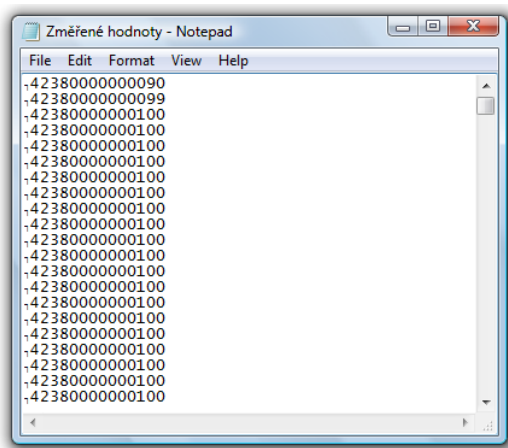
7.1.2 RS-232

Pro komunikaci po sériové lince je použita sada instrukčních bloků s názvem *VISA* (The Virtual Instrument Software Architecture). Prvním krokem je zapotřebí sériovou komunikaci nastavit, to zajišťuje blok s názvem *VISA Configure Serial Port*. Pomocí *VISA resource name* je možné vybrat správný port, na kterém probíhá komunikace s PC. Rychlost komunikace je defaultně nastavena na 9600 baud. Dalším krokem je *VISA Set I/O Buffer Size* pro nastavení vyrovnávací paměti a zabezpečení plynulosti komunikace. Následuje blok *VISA Read*, který z komunikace čte šestnácti bytové balíčky data (viz Obr. 18).



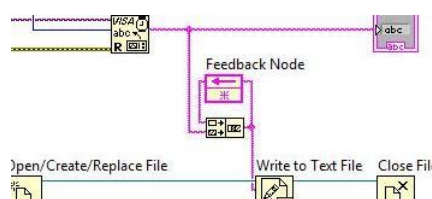
Obr. 18: RS-232, Block Diagram

Paralelně s tímto přenosem dat probíhá jejich ukládání. Blok *Open/Create/Replace File* pro otevření, vytvoření nebo nahrazení souboru umožňuje se zvoleným souborem dále pracovat. Po zadání cesty se soubor otevře nebo vytvoří nový. Následně se do něj přijatá hodnota запиše pomocí *Write To Text File* a soubor se uzavře blokem *Close File*. Blok *Elapsed Time* zpomaluje krok čítání a ukládání na 1 vteřinu. Uložený textový soubor obsahuje pouze seznam 16 místných textových řetězců, jak je ukázáno na Obr. 19.



Obr. 19: RS-232, Záloha hodnot

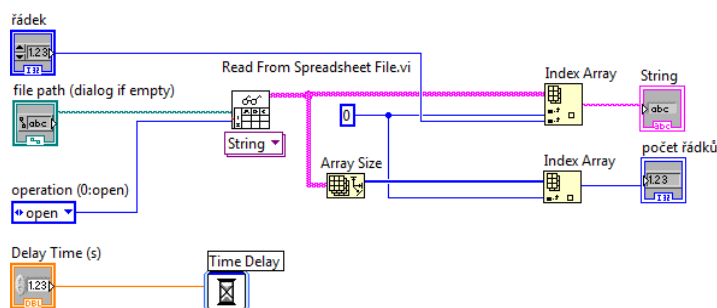
Ukládání do souboru je blíže zobrazeno na Obr. 20. Textový řetězec vstupuje do bloku, který ho zařadí na další řádek, za předchozí hodnoty. Tento celý svazek se každou vteřinou uloží.



Obr. 20: RS-232, Sloupec řetězců

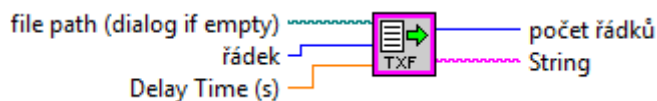
7.1.3 Ze souboru

Mód pracuje s uloženými daty módem RS-232, jak je ukázáno na Obr. 19. Tyto data vnímá jako tabulku o jednom sloupci a několika řádcích. Po vybrání textového souboru či zadáním cesty do *file path* dojde k jeho otevření pomocí *Read From Spreadsheet file.vi*. Dále se pracuje pouze s polem o šířce jednoho sloupce a o délce přímo odpovídající počtu přijatých datových balíčků od wattmetru v módu RS-232. Pro výběr konkrétního řádku se používá indexování blokem *Index Array*, do kterého vstupuje pole, o již zmíněných rozměrech, spolu se dvěma hodnotami. První *index 0* určuje, jaký řádek se má zobrazit. Druhý *index $n-1$* zobrazuje, od jakého řádku se má začít, tento vstup je nastaven na hodnotu 0. *Index Array* vyjme požadovaný řádek a zobrazí v *String Indicator*. Druhý *Index Array* propojený s blokem *počet řádků* pouze informuje, kolik změřených hodnot máme k dispozici. Poslední částí je ovladač *Delay Time*, kterým je možno měnit velikost zpoždění na bloku *Time Delay* a tím i rychlost čtení dat. Podobně jako zpoždění v módu Simulace. Ukázka této části programu na Obr. 21.



Obr. 21: Ze souboru, Block Diagram

SubVI (viz Obr. 22) je vytvořeno tak, že vstupem je *file path* pro zadání cesty k souboru, *řádek*, který umožňuje uživateli načíst hodnotu libovolné řádku a *Delay Time* pro regulaci zpoždění. Na výstupu jsou pouze 2 signály, *počet řádků* pro přibližnou orientaci v otevřeném souboru a *String* nesoucí textový řetězec určený hodnotou vstupu *řádek*.

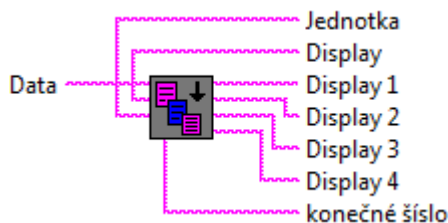


Obr. 22: SubVI Ze souboru

7.2 Rozšifrování dat

Jeden z nestěžejnějších SubVI je *rozšifrování dat*. Jak je vidět na Obr. 23 jediným vstupem je datové slovo pod názvem *Data*, které je získáno jedním z volitelných módů (kapitola Volitelné módy). Výstupů má tento blok ale mnoho, jsou to některé z informací, které lze z datového slova získat, plus parametry z nich stvořené potřebné k dalšímu běhu programu a správnému zobrazení.

- *Jednotka*
Číselný kód pro určení jednotky měřené veličiny.
- *Display*
Číslo od 1 do 4 rozhodující, na kterém displeji má být informace zobrazena.
- *Konečné číslo*
Konečná podoba čísla obsahující i polaritu a desetinnou čárku. Ne však jednotku.
- *Display 1 až Display 4*
 - Tyto 4 výstupy nesou již konečnou podobu zobrazovaného čísla s polaritou, desetinnou čárkou a jednotkou veličiny.

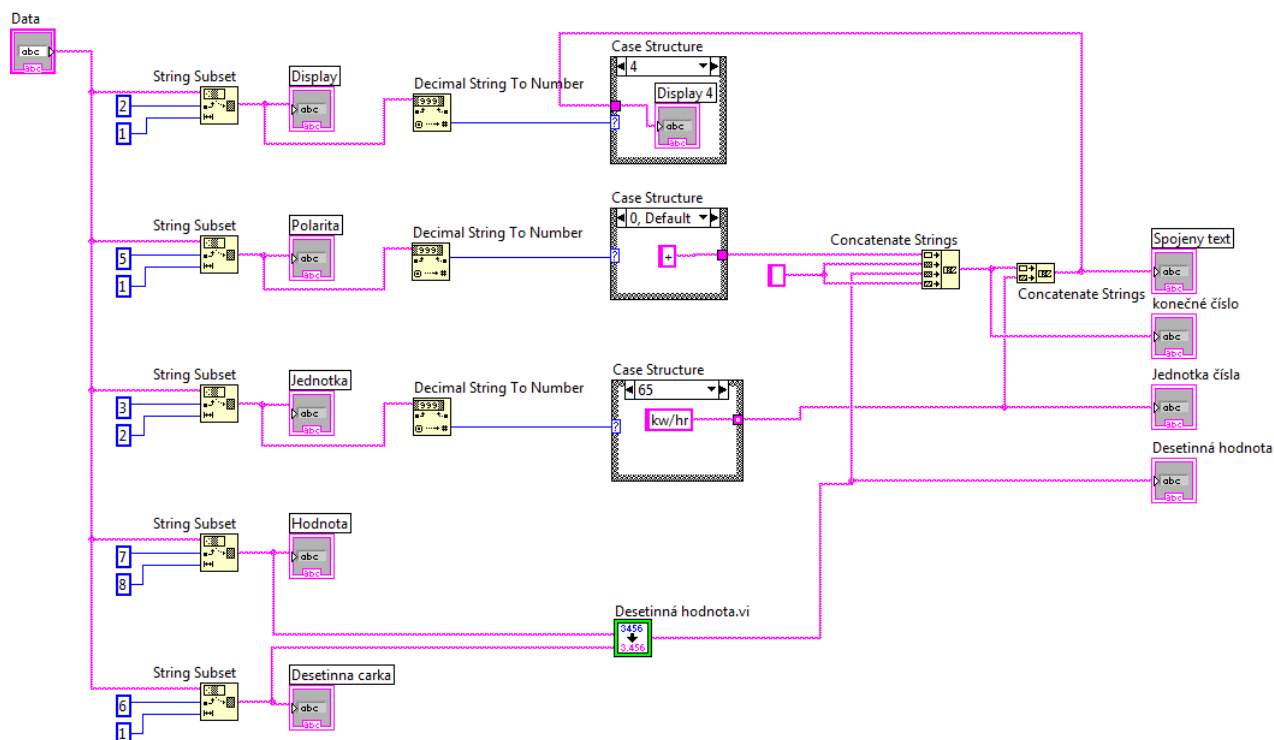


Obr. 23: SubVI Rozšifrování dat

Jelikož je textový řetězec vždy stejně dlouhý, je možné separovat konkrétní hodnoty pomocí *String Subset*. Vstupem do tohoto bloku je textový řetězec a další dva parametry. *Offset*, jehož hodnota říká od kolikátého znaku se má separace začít, a *length*, který určuje kolik znaků se bude separovat.

Př.: Číslo, určující jaký displej bude danou hodnotu zobrazovat, je udáváno v datovém slově na 2. místě. *Offset* tedy nastavíme na číslo 2. Jelikož požadována informace má velikost jednoho znaku, *length* nastavíme na hodnotu 1. Na výstupu z bloku *String Subset* je tedy jednomístný znak z druhé pozice datového slova. Konkrétní pozice informací je podrobně vysvětleno v kapitole Komunikace.

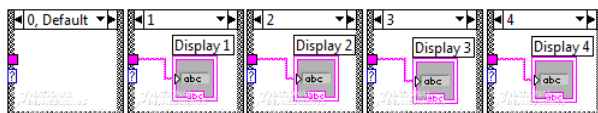
Na tomto principu je separována každá informace nesená datovým slovem. Konkrétní část programu uvedena na Obr. 24. Získané hodnoty jsou ale pro uživatele nic neříkající. Nadále jsou používány jako parametry pro odvození zobrazitelných informací. *Display*, *Polarita* a *Jednotka* jsou pomocí funkčního bloku *Decimal String To Number* převedeny z formátu *string* do formátu *integer*. Díky tomu lze těmito hodnotami inicializovat smyčky *Case* a určit tak jiným parametrům konkrétní znaky, nebo řídit zobrazení displejů, jako v případě informace *Display*.



Obr. 24: Rozšifrování dat, Block Diagram

- *Display*

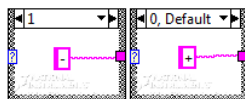
S jeho pomocí je konečná podoba měřené veličiny zobrazována správným displejem, Obr. 25.



Obr. 25: Rozšifrování dat, Case, Display

- *Polarita*

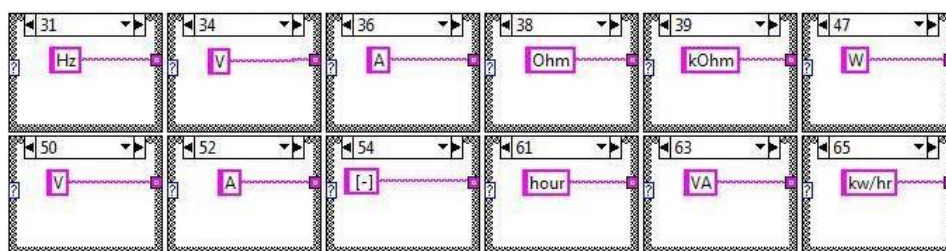
Pokud je na selektor smyčky *Case* přivedeno číslo 1 (resp. 0), tak se před změřenou hodnotu předradí znaménko mínus (resp. plus), Obr. 26.



Obr. 26: Rozšifrování dat, Case, Polarita

- *Jednotka*

Stejným způsobem, jako v předešlých dvou případech, se k číslu přiřadí i jednotka. Která číselná hodnota odpovídá jaké jednotce veličiny je určeno výrobcem a zobrazeno na Obr. 27.



Obr. 27: Rozšifrování dat, Case, Jednotka

- *Hodnota a Desetinná čárka*

Tyto dva výstupy jsou pouze využity jako vstupy do SubVI *Desetinná Hodnota*. Kde je s nimi dále pracováno.

7.2.1 Desetinná hodnota

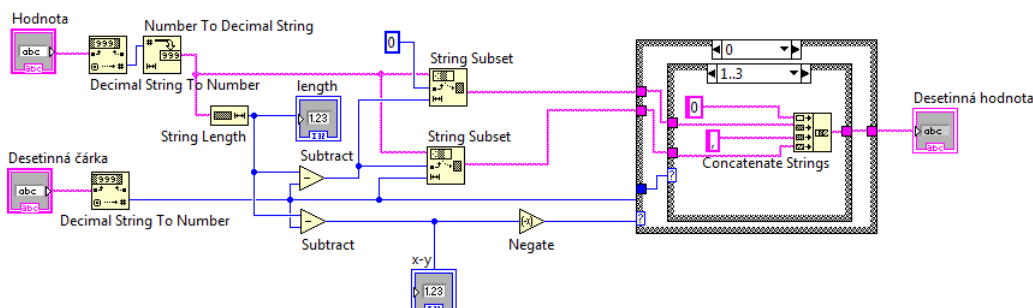
Úkolem tohoto SubVI, Obr. 28, je zařadit desetinnou čárku na správnou pozici změřené veličiny. Vstupem je zde *Desetinná čárka*, jejíž hodnota určuje pozici desetinné čárky a *Hodnota* obsahující naměřenou hodnotu bez desetinné čárky. Výstupem pak je *Desetinná hodnota* zobrazující číslo ve správném tvaru a formátu.



Obr. 28: SubVI Desetinná hodnota

Princip činnosti je takový, že zbavíme vstupu *Hodnota* všech nadbytečných znaků „0“, kterými byl doplněn na osm pozic. Dále podle velikosti vstupu *Desetinná čárka* je odděleno příslušné množství znaků od *Hodnota*, a to z pravé strany. Dojde tak k rozdělení vstupu *Hodnota* na 2 části, mezi které se vloží znak čárky. Následné sloučení zajišťuje blok *Concatenate Strings*. Kolize nastává v případě, že vstup *Hodnota* po odstranění nul má méně znaků než je pozice desetinné čárky, jinak řečeno má být zobrazeno číslo menší než 1. Pro

takový případ je zde kombinace dvou smyček *Case*, reagujících na rozdíl počtu znaků vstupu *Hodnota* a čísla na vstupu *Desetinná čárka*. Slučování znaků už pak není tak jednoznačné a je potřeba o doplnění nuly před a o potřebný počet nul za desetinnou čárkou. Celé propojení na Obr. 29.

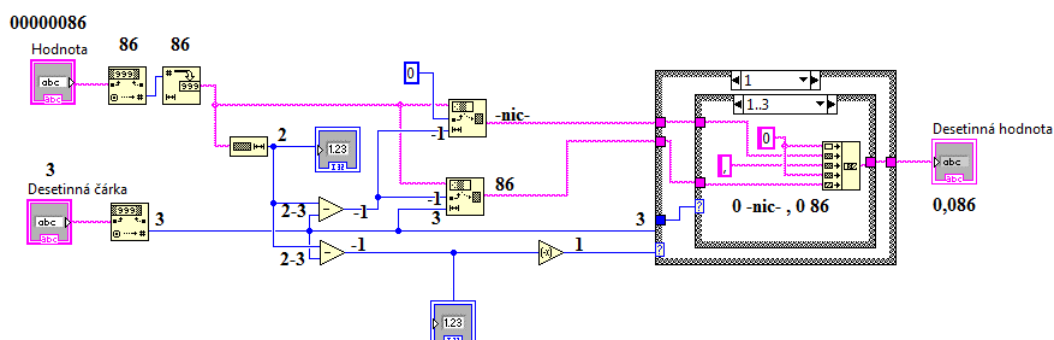


Obr. 29: Desetinná hodnota, Block Diagram

Pro odstranění předřadných nul vstupu *Hodnota* je využito bloku *Decimal String To Number*. Protože je ale zapotřebí pracovat se znaky a ne s čísly, je nutné převést *Hodnotu* z datového typu *integer* zpět na *string* blokem *Number To Decimal String*. Takto očesaný signál postupuje do dvou bloků *String Subset*. *Offset* prvního *String Subset*, který má na výstupu levou část vstupu *Hodnota*, je nastaven na nulu. *Length* je určen rozdílem počtu znaků *Hodnota* a velikostí čísla *Desetinná čárka*. Tento *Length* je poté použit jako *offset* druhého *String Subset* nesoucí pravou část. *Length* druhého *String Subset* je nastaven velikostí čísla *Desetinná čárka*.

Příklad výpočtu

Na vstupu *Hodnota* je přivedeno 00000086 a na vstupu *Desetinná čárka* číslo 3. Tímto je jasné řečeno, že před třetím znakem zprava má být umístěna desetinná čárka. Výslednými znaky na výstupu *Desetinný hodnota* by tedy mělo být 0,086. Postup je zobrazen na Obr. 30.



Obr. 30: Desetinná hodnota, Block Diagram, náhled na výpočet

Hlavní smyčka *Case* je řízena negovaným rozdílem počtu znaků vstupu *Hodnota* a velikosti čísla vstupu *Desetinná hodnota*. Má 4 stavy: *Default*; 0; 1 a 2. V každém ze čtyř stavů obsahuje smyčku vedlejší, která je inicializována podle velikosti vstupu *Desetinná hodnota*. Taková smyčka nabývá pouze 2 stavů: 0, *Default* a 1..3. Možných kombinací, které mohou nastat je 8. Avšak veškeré kombinace, ve kterých je vedlejší smyčka nastavovaná nulou, probíhají stejně. V takovém případě nebylo naměřeno desetinné číslo a vstup *Hodnota* pouze projde smyčkami na výstup. Ve zbývajících čtyřech smyčkách, už dochází ke slučování s dalšími znaky. Smyčky vyobrazeny na Obr. 31.

- 0; 1..3

0 na první smyčce naznačuje, že počet znaků a pozice desetinné čárky si odpovídají. 1..3 na smyčce druhé pak říká, že desetinná pozice je nenulová. Výstupem ze smyček bude složený znak: 0,X (kde X nahrazuje změřené číslo).

- default; 1..3

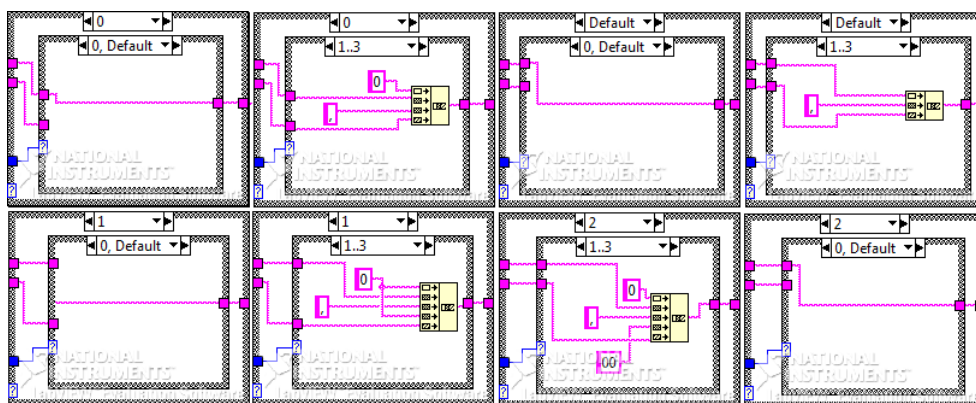
Default je nastaven vždy když přijde hodnota, která neodpovídá intervalu <0;2>. Číslo tedy není menší než nula a výstup bude vypadat takto: X₁,X₂ (kde X₁ je levá polovina a X₂ pravá).

- 1; 1..3

Výstupu 0,0X (kde X nahrazuje změřené číslo).

- 2; 1..3

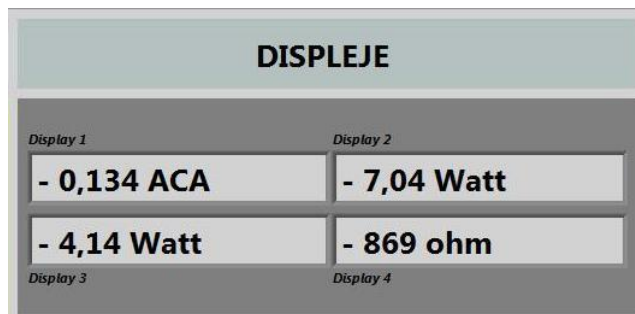
Výstupu 0,00X (kde X nahrazuje změřené číslo).



Obr. 31: Desetinná hodnota, Case, desetinná čárka

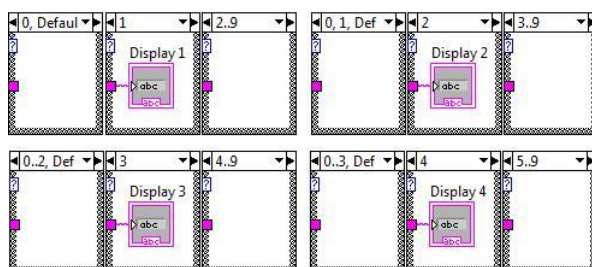
7.3 Zobrazení na čtyřech displejích

Wattmetr je vybaven čtyřmi displeji, stejně tak je využito i v programu. Ukázka na Obr. 32.



Obr. 32: Zobrazení na čtyřech displejích, Front Panel

Ke každému displeji vede jeden výstup z SubVI *rozšířování dat* se stejným názvem (tj. *Display 2* je připojen k výstupu s názvem *Display 2*). Tím dochází k zobrazování konečné hodnoty se znaménkem i jednotkou, ve které byla změřena, na správný displej. Displeje jsou aktualizovány periodicky. Pokud není displej zrovna aktualizován, neobsahuje žádnou hodnotu. Aby hodnoty na displejích neproblikávaly, je každý displej umístěn ve smyčce *Case*, která povolí zápis pouze ve správný okamžik. Možné stavy smyček *Case* jsou zobrazeny na Obr. 33.



Obr. 33: Zobrazení na čtyřech displejích, Case, Display 1-4

Display 1 až *Display 4* je možno uvést do 2 stavů. Prvním stavem je defaultně nastaveno zobrazení displeje a druhým jeho nezobrazení. V druhém případě dochází k neukládání jeho hodnoty do tabulky.

7.4 Správa programu

Správou programu je z pohledu uživatele několik ovládacích prvků, kterými nastavuje vlastnosti čítání hodnot. Každý mód používá jiné prvky. Pro zobrazení jednotlivých ovládacích prvků je využito vlastnosti *Visible*. Podle zvoleného módu se do této vlastnosti zapíše hodnota *True* pro zobrazení prvku, popřípadě hodnota *False* pro jeho nezobrazení.

Princip spočívá v umístění bloků *Property Node*, které umožňují ovlivnit vlastnost *Visible*, mimo smyčku *Case*. Touto smyčkou se volí mód a spolu s ním se mění konstanta *Boolean Constant* (v pravém dolním rohu smyčky), která pak ovlivňuje bloky *Property Node*.

7.4.1 Simulace

Při simulaci má uživatel umožněno pouze měnit rychlost, se kterou se data zobrazují. Ovládání je uskutečněno posuvným jezdce s logaritmickým měřítkem. Část *Front Panel* na Obr. 34.

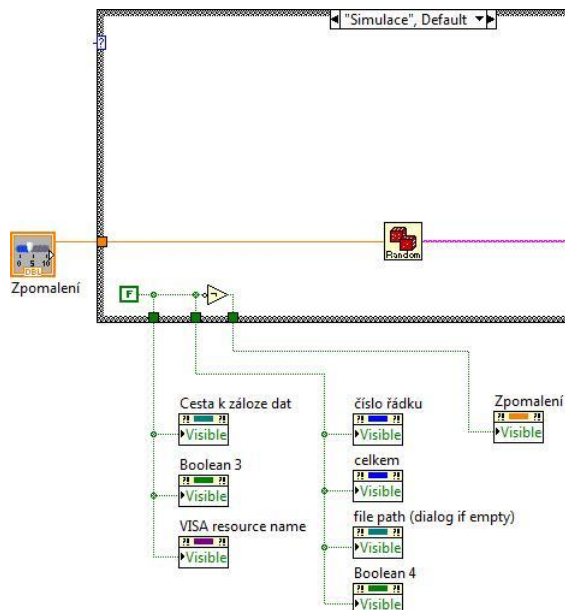


Obr. 34: Simulace, Správa

V tomto módu se zapisuje do vlastnosti *Visible* hodnota *True* ale pouze pro posuvník rychlosti generování, Obr. 35. Tento *Property Node* je pojmenovaný jako *Zpoždění*

7.4.2 Ze souboru

Uživatelský mód *Ze souboru* má pole *SPRÁVA* s pěti prvky, Obr. 36. Jelikož se jedná pouze o jakousi rekonstrukci proběhnutého měření, je umožněno měnit rychlost čítání dat ze souboru. Dva indikátory pod posuvníkem rychlosti informují uživatele o pořadí aktuální hodnoty a o množství hodnot, které jsou k dispozici. Následující textové pole je prostor pro vyhledání/zadání cesty k souboru, ze kterého bude program čerpat. Nejnižší je umístěno tlačítko, které spustí čítání. Do všech *Property Node* těchto prvků je zapsána hodnota *True*.

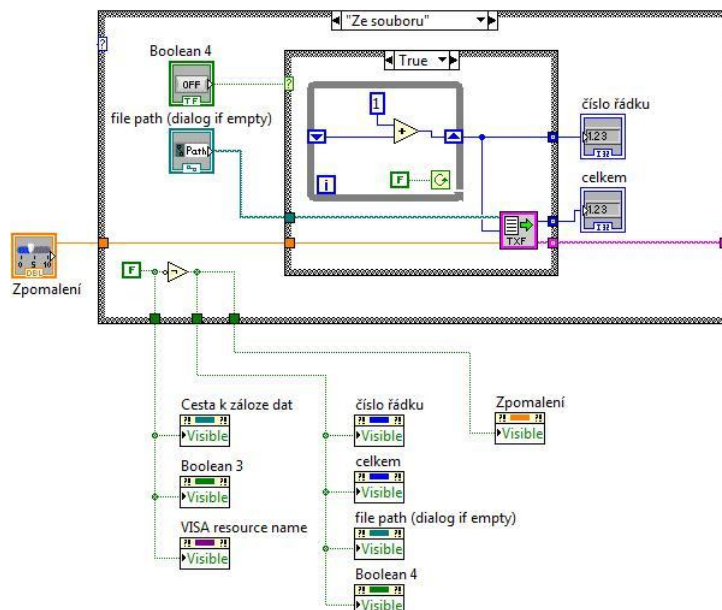


Obr. 35: Simulace, Block Diagram



Obr. 36: Ze souboru, Správa

V případě zpomalování je tato problematika řešena stejně jako u módu *Simulace*. Počet zálohovaných hodnot je indikován pomocí bloku *Number Indicator* a je vyveden jako výstup ze SubVI *Ze souboru*. Aktuální číslo načteného řádku je zobrazeno pomocí stejného bloku *Number Indicator*. Tato hodnota se inkrementuje s každým proběhnutým cyklem. O to se stará smyčka *While loop* a posuvný registr *Shift Register*, k jehož hodnotě se přičítá jednička. Tato hodnota je informací pro uživatele, ale také vstupem do SubVI *Ze souboru* a určuje, jaký řádek se zobrazí. Následuje textové pole, do kterého je možno přímo napsat cestu ke zdrojovému souboru. Popřípadě kliknout na ikonu složky a cestu dohledat. Tlačítkem *Start* se spustí čítání hodnot. Toto tlačítko je označeno jako *Boolean 4*. Ukázka programu je uvedena na Obr. 37.



Smyčky *Case* je využito pro spuštění běhu programu pouze po stisknutí tlačítka *Start*. V případě opačné inicializace, tedy v případě *False*, neproběhne nic. Zobrazení stavů na Obr. 38.

7.4.3 RS-232

SPRÁVA

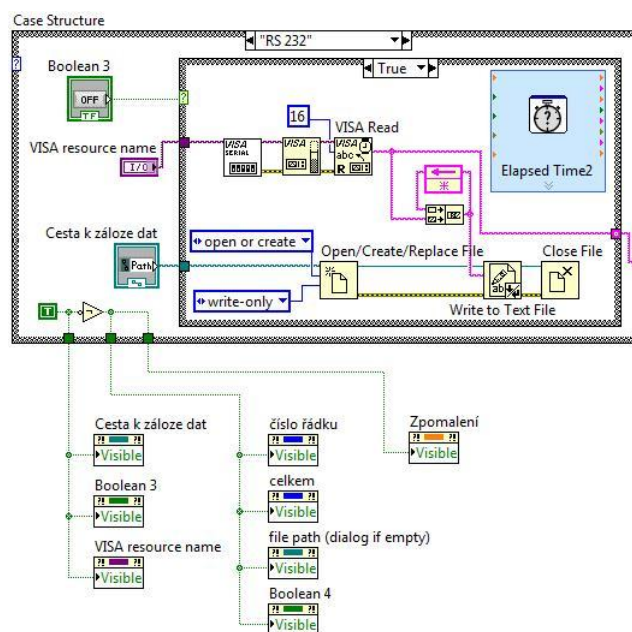
VISA resource name

Cesta k záložce dat

Start

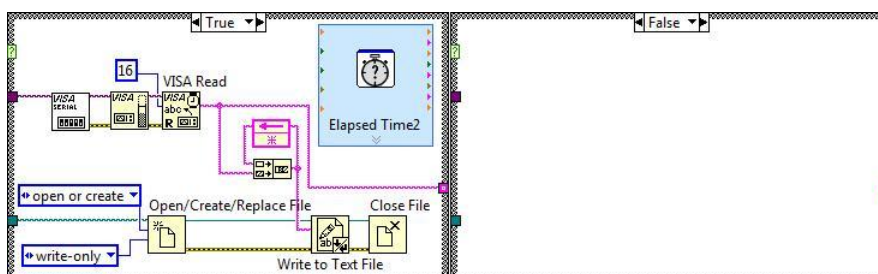
Kliknutím na šipku u *VISA resource name* se otevře roletové menu, které nabídne komunikační porty. Uživatel si sám zvolí port, ke kterému je wattmetr připojen. Jako cestu k záloze dat je možno zadat cestu k neexistujícímu souboru. V takovou chvíli je podmínkou doplnění jména, pod kterým bude soubor vytvořen. Druhou možností je zadat cestu k již existujícímu souboru. Tlačítkem *Start* se program spustí.

Konstanta *True* je zapsána do *Property Node* prvků *Cesta k záloze*, *VISA resource name* a *Boolean 3*. Negovaná hodnota této konstanty je poté zapsána do *Property Node* ostatních prvků. Tím je zabezpečeno zobrazení pouze odpovídajících funkcí. Viz Obr. 40.



Obr. 40: RS-232, Block Diagram

Tlačítko *Start*, realizované blokem *Boolean 3*, inicializuje smyčku *Case*. Tlačítko zabezpečuje čítání hodnot až po nastavení parametrů. Na Obr. 41 jsou zobrazeny 2 možné stavy smyčky *Case*.

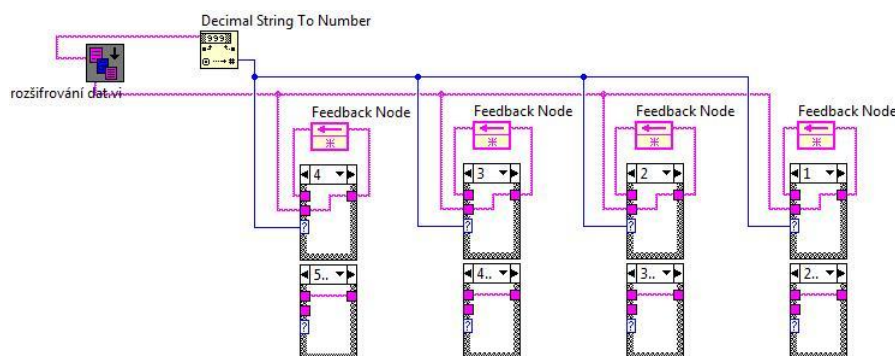


Obr. 41: RS-232, Case

7.5 Sběr dat

Hodnoty zobrazované na displejích má uživatel možnost ukládat do souboru, který je následně schopný otevřít v tabulkovém editoru. O tom, který displej bude a nebude ukládán, má uživatel možnost rozhodovat sadou přepínačů.

Displeje obsahují hodnotu spolu s její jednotkou, což by v tabulce neumožňovalo efektivně hodnoty využít. Pro další práci s hodnotami je tedy důležité, aby se s nimi dalo pracovat, jako s čísly. Ukládaná hodnota tedy neobsahuje jednotku měřené veličiny. Takto upravená hodnota se ukrývá na výstupu z SubVI *rozšifrování dat* pod jménem *konečné číslo*. I když je tento výstup ve formátu *string*, uložená hodnota v tabulce obsahuje pouze znaky, které tabulkový editor hodnotí jako číslo. O rozčlenění výstupu na hodnoty pro konkrétní displej se starají čtyři smyčky *Case*. Do každé vstupuje *konečné číslo*, ale smyčkou prochází pouze číslo pro konkrétní displej. V případě, že hodnota není v daný moment aktualizována, je využito bloku *Feedback Node*. Blok funguje jak zpětná vazba, která uchovává hodnotu naposledy aktualizovanou. Smyčky a jejich stavy jsou ukázány na Obr. 42.



Obr. 42: Sběr dat, aktualizace hodnot + Case

Každá ze smyček je inicializována číslem displeje, které je získáno blokem *Decimal String To Number* z výstupu *display*. Tímto vznikají z jednoho výstupu *konečné číslo* čtyři datové spoje, které se dále využívají všichni stejným způsobem a však každý zvlášť.

7.5.1 Nastavení sběru dat

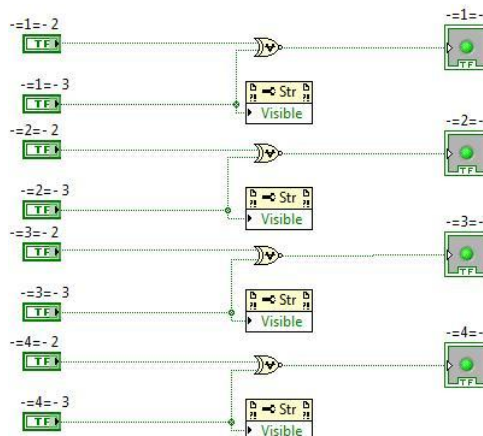
Pomocí několika přepínačů je umožněno ovlivňovat ukládání hodnot do tabulky. Pro každý displej jsou v poli *Nastavení sběru dat* umístěny 2 přepínače (viz Obr. 43). Horním přepínačem je možno určit zda displej bude zobrazen. Za předpokladu, že nebude displej zobrazován, nebude poté ani ukládán do tabulky. Spodním přepínačem je možné negovat nastavení horního přepínače, ale pouze v oblasti ukládání či neukládání hodnoty do tabulky.

Pokud je horní přepínač nastaven pro nezobrazení displeje a neuložení hodnoty do tabulky, může uživatel spodním přepínačem ovlivnit, zda hodnota nezobrazovaného displeje bude i přesto ukládána do tabulky či nikoli. Logika přepínačů je zastoupena v *Block Diagram* negovaným exkluzivním součtem. Pro přehlednost, zda se hodnota bude či nebude ukládat, je v řádku *Indikátor uložení* umístěna čtveřice diod. Diody svítí pouze v případě, že se umožňuje ukládání tabulky.



Obr. 43: Nastavení sběru dat, Front Panel

Přepínače s indexem 2 (např.: $\text{--}3\text{--}$ 2) se na *Front Panel* nacházejí v řádku *zobrazení*. Na každý z přepínačů je napojen *Property Node* s názvem *Visible* pro příslušný displej. Je tak umožněno nastavování displeje se stejným číslem (např.: $\text{--}3\text{--}$ 2 zobrazuje/nezobrazuje *Display 3*). Obdobně přepínače s indexem 3 ovlivňující pouze konkrétní sloupce v tabulce.

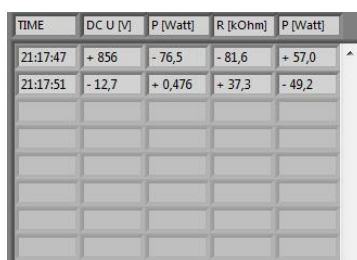


Obr. 44: Nastavení sběru dat, zakázání uložení

Signály obou tlačítek jsou vedeny do bloku *Not Exclusive Or*. Dioda na jeho výstupu se tedy rozsvítí pouze v případě, že oba signály nabývají hodnotu *log. 1* nebo *log. 0*. Zapojení je na Obr. 44.

7.5.2 Tabulka

Tabulka je složena z pěti sloupců, kde v prvním se zobrazuje čas, zbylé čtyři sloupce odpovídají každý jednomu z displejů. Popis je k Obr. 45. Každý ze sloupců začíná hlavičkou a nemá předem určený počet řádků. Hlavička zobrazuje měřenou veličinu spolu s její jednotkou, kromě prvního sloupce. Je realizována zobrazovači.

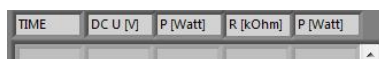


TIME	DC U [V]	P [Watt]	R [kOhm]	P [Watt]
21:17:47	+ 856	- 76,5	- 81,6	+ 57,0
21:17:51	- 12,7	+ 0,476	+ 37,3	- 49,2

Obr. 45: Tabulka, Front Panel

7.5.3 Hlavička tabulky

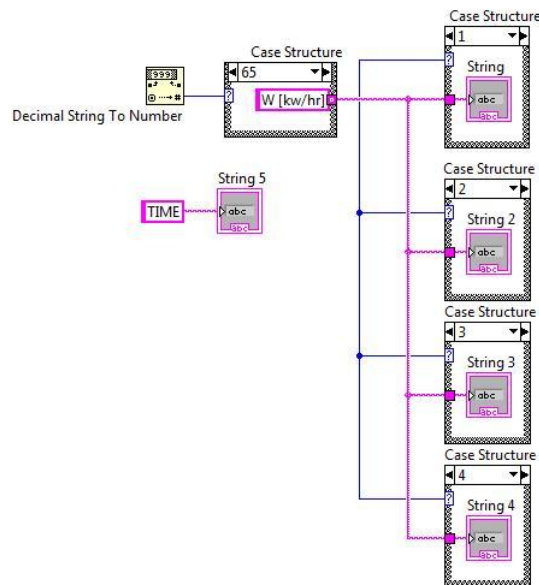
Hlavička je vytvořena pomocí pěti na sobě nezávislých zobrazovačů *String* až *String 5* na Obr. 46.



TIME	DC U [V]	P [Watt]	R [kOhm]	P [Watt]
------	----------	----------	----------	----------

Obr. 46: Hlavička tabulky, Front Panel

První zobrazovač je naplněn konstantou *TIME*. Obsah ostatních zobrazovačů je vytvářen na podobném principu jako jednotka veličiny v kapitole Rozšifrování dat. Řídícím signálem je *Jednotka*, výstup ze SubVI *Rozšifrovaná data*. Blokem *Decimal String To Number* je tento signál konvertován do formátu *integer* a tím je následně inicializována smyčka *Case*. Podle hodnoty signálu bude zobrazována veličina spolu s jednotkou. O tom, který zobrazovač bude v danou chvíli aktualizován, rozhoduje stejný signál jako v případě zobrazování změřených hodnot na displejích. I v tomto případě signál inicializuje čtyři smyčky *Case*, kde každá obsahuje jeden ze zobrazovačů. Schéma je na Obr. 47.



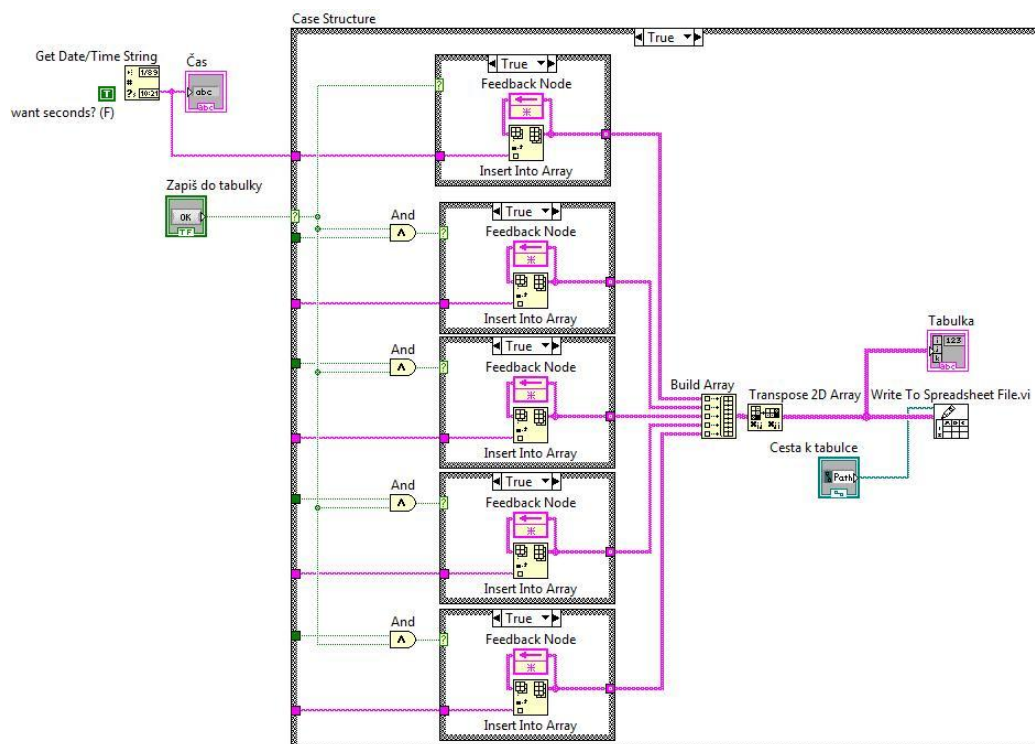
Obr. 47: Hlavička tabulky, Block Diagram

7.5.4 Vložení do tabulky

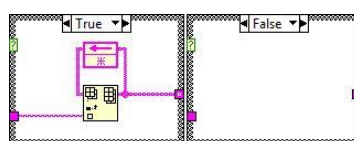
K vložení do tabulky dochází stisknutím tlačítka *Zapiš do tabulky*. Tlačítko ovládá smyčku *Case*, která proběhne pouze při hodnotě *True*. Uvnitř této smyčky se nachází dalších pět smyček typu *Case*, které zabezpečují také to, aby byla hodnota do tabulky vložena v případě, že uživatel uložení nezakázal. Do každé smyčky vstupuje pouze signál určený pro konkrétní sloupec v tabulce, neboli display. Pomocí bloku *Insert into array* je vložena hodnota do pole, které je pomocí bloků *Feedback Node* uchováváno do jeho další aktualizace. Při každém stisknutí tlačítka *zapiš do tabulky* se aktuální hodnota přiřadí k již uloženým hodnotám v poli jako následující sloupec. V tuto chvíli vzniká pět polí, každé o jednom řádku a takovém počtu sloupců, kolikrát bylo stisknuto tlačítko pro zápis. Pro každý displej a časový údaj je přidělen jeden řádek. Blokem *Build Array* jsou tyto pole sloučena do jednoho a je vytvořeno pole o pěti řádcích a x sloupcích. Pro zobrazení hodnot v přijatelnějším formátu, tedy v řádcích pod sebou, je nutné pole transponovat blokem *Transpose 2D Array*. Kompletní schéma zapojení je na obrázku Obr. 48.

První sloupec tabulky určující čas pořízení hodnot je zobrazen pomocí bloku *Get Date/Time String*. Na vstup *want sekund?(F)* je zapsána hodnota *True* pro zobrazení času ve formátu *hh:mm:ss*. Výstup *Time* je také přiveden na blok *String Indicator* s názvem *Čas*.

Možné stavy vnitřních smyček *Case* jsou pouze *True* nebo *False*, neboli vykoněj či nedělej nic. Na Obr. 49 jsou vyobrazeny stavy, které smyčky nabývají.



Obr. 48: Vložení do tabulky, Block Diagram



Obr. 49: Vložení do tabulky, Case, jednotlivá pole

7.5.5 Uložení do tabulky

Pro ukládání do tabulky je nutné zadat cestu, kde se tabulka bude nalézat. Je možné název souboru napsat i s příponou konkrétního tabulkového editoru. Soubor se při prvním stisknutí tlačítka *Zapiš do tabulky* vytvoří. Vše je vidět na Obr. 50. Při výběru již vytvořené tabulky se však hodnoty nepřirazují, nýbrž se vytváří tabulka nová pod stejným jménem, která nahrazuje předchozí soubor.



Obr. 50: Uložení do tabulky, Front Panel

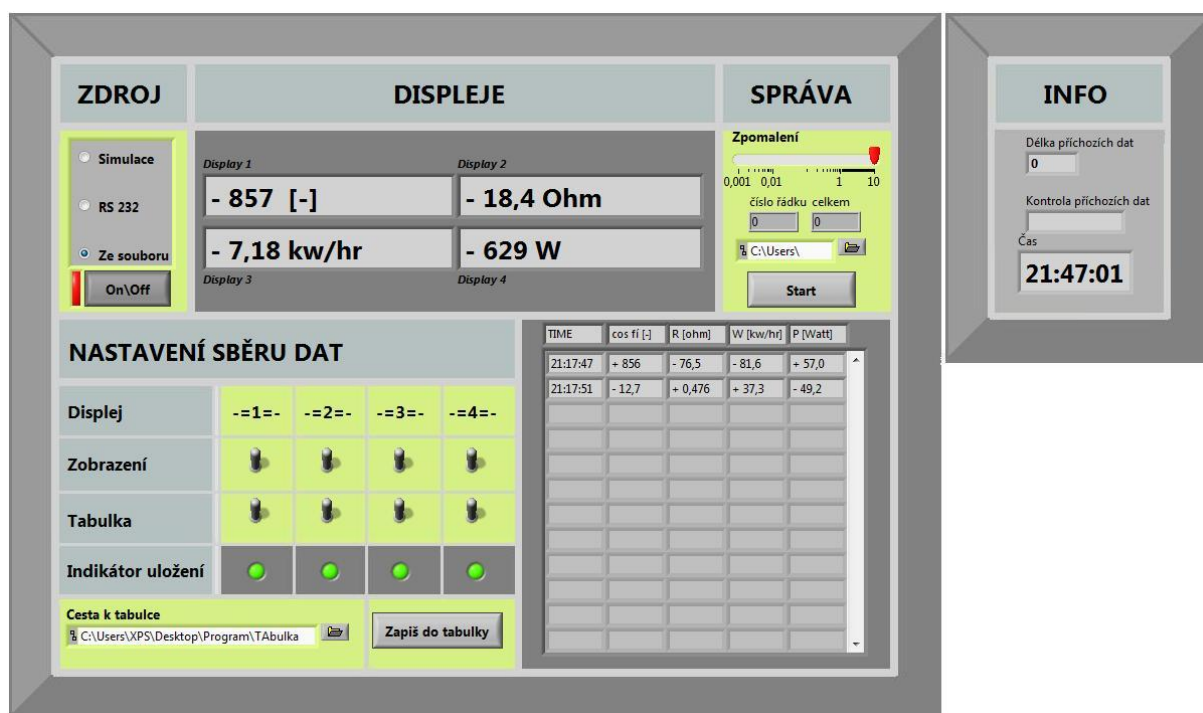
Pole s hodnotami je ukládáno pomocí bloku *Write To Spreadsheet File*, do kterého vstupují 2 signály. Prvním je informace, kde bude tabulka vytvořena, zadávána do terminálu *Cesta k tabulce*. Druhým pak samotné 2D pole hodnot. Toto zapojení je součástí Obr. 48.

8 Přehled pro uživatele

Program, jako všechny aplikace vytvořené v prostředí LabView, tvoří Front Panel a Block Diagram. Front panel je barevným pozadím členěn vždy a uživatel pracuje pouze s ním. Block Diagram je barevně rozlišen pouze v této kapitole pro jeho přestavení.

8.1 Front panel

Celý ovládací panel kromě rámečku je barevným pozadím rozdělen do čtyř částí. Světle modrá barva uživatele navádí a jsou v ní napsány názvy polí. Světle zelená barva určuje místa, ve kterých uživatel zadává a mění parametry či jinak obsluhuje program. Tmavě šedivé pole je místo, kde jsou zobrazovány hodnoty a světle šedivé pozadí obsahuje pouze informaci o přijatých datech a čase. Konečná podoba programu, se kterou bude uživatel pracovat, je na Obr. 51. Hodnoty byly generovány náhodně.

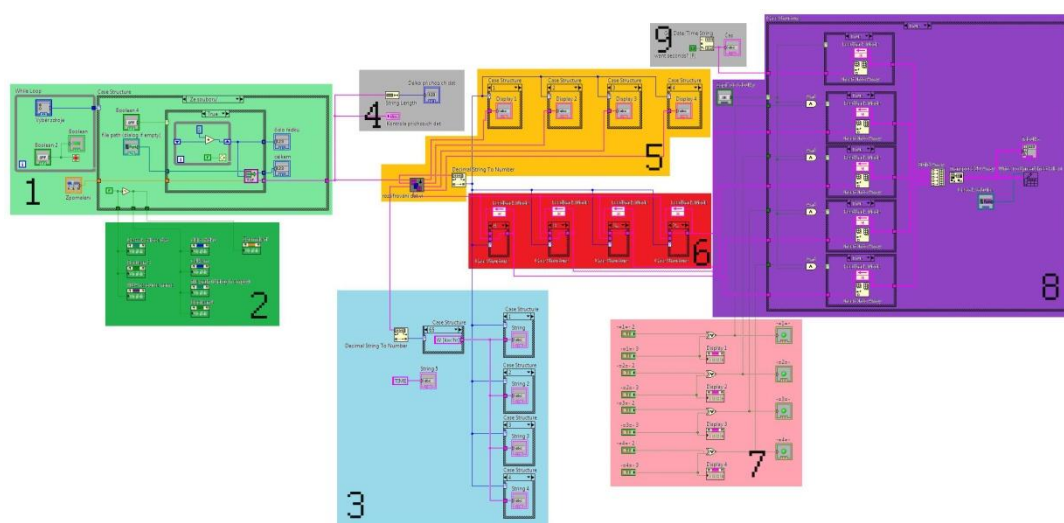


Obr. 51: Front Panel

Pro ovládání uživatel postupuje z levého horního rohu do pravého dolního rohu. V první řadě vybere zdroj, následně nastaví parametry v poli *SPRÁVA* (tj. např. rychlost, vybrání komunikačního portu, zapnutí komunikace, atd.). Pokud chce uživatel hodnoty i zaznamenávat, vybere si v *NASTAVENÍ SBĚRU DAT* hodnoty, které nejsou žádoucí. Přejde k zadání cesty k tabulce. Po jejím zadání už jenom stiskem *Zapiš do tabulky* určuje, v jaký moment budou zvolené hodnoty uloženy.

8.2 Vnitřní struktura

Tento bod pouze orientačně uživateli nastiňuje chod programu a to na Obr. 52. Ve světle zeleném poli, též pod číslem 1, probíhá volba módu. V tmavějším odstínu zelené se nachází zobrazení konkrétních nástrojů pro správu, číslo 2. V šedivě zbarvených obdélnících s číslem 4 se zajišťuje informace o přijatém datovém slově a s číslem 9 čas. Žlutooranžová barva s číslem 5 pokrývá část programu, která zajišťuje rozšifrování přijatého slova a jeho zobrazení na displejích. Červené pole s číslem 6, a následně fialové pole s číslem 8, zajišťuje přípravu a ukládání hodnot do tabulky. Pole o růžové barvě a číslem 7 ohraničuje bloky pro volbu zakázání uložení do tabulky. Modré pozadí číslo 3 pokrývá zobrazení hlavičky tabulky.



Obr. 52: Block Diagram

9 Vzdálený přístup

Vzdálený přístup je společností National Instrument plně podporován. Jednou z funkcí programu LabView je tzv. *Remote Panel*. Výhodou takového *Remote Panel* je možnost ovládat a kontrolovat pracoviště, které může být umístěno v podstatě kdekoli. Dále je pod označením PC-1 a pracoviště, ze kterého je ovládáno zkratkou PC-2. Vzdálená pracoviště je možné propojit dvěma způsoby.

- Připojení *Remote Panelu* v LabView
- Otevření odkazu ve webovém prohlížeči

V jednodušším případě, a to v navázání spojení webovým prohlížečem, se na uživatele kladou pouze dva nároky:

- Vzdáleně řízené PC-1 musí mít nainstalováno grafické programovací prostředí LabView
- PC-2 pro vzdálené řízení musí obsahovat tzv. *Run-Time Engine*, a to v souhlasné verzi s LabView na ovládaném PC-1

V druhém případě je nutné znát IP adresu PC-1 a přesný název VI, které na něm uživatel chce ovládat.

Pokud je jedna z těchto cest zabezpečená je možné pracoviště propojit a to třemi způsoby LAN, WAN a pomocí internetu.

9.1 Očekávání

Požadavkem na vzdálené ovládání bylo, aby uživatel mohl plně používat vytvořený program, ale zároveň v něm nemohl provádět změny. Jinak řečeno, uživatel by mohl nastavovat a ovládat vše co je součástí *Front panel* na Obr. 51 s. 49. Vnitřní propojení a struktura programu na Obr. 52 by mu zůstala utajena. Dalším požadavkem bylo úplné přesunutí pracoviště. Aby nedocházelo k pouhému sledování či ovládání měření ale také možnost dostat se k uloženým datům.

V prvním případě je na výběr, zda uživatel bude moci měření pouze sledovat, nebo i ovládat. Nikdy ale nebude mít možnost zasahovat do struktur programu. V případě druhém a to v možnosti, kdy je povoleno program vzdáleně ovládat (nejenom sledovat) je automaticky určeno jako úložiště dat nejenom pracoviště PC-1 ale zároveň PC-2. Na obou PC

je tedy souhlasně tvořen soubor s názvem tabulky či zálohy. Uživateli při pokusu nastavení cesty k zálohám bude defaultně nabídnuta cesta v PC-2.

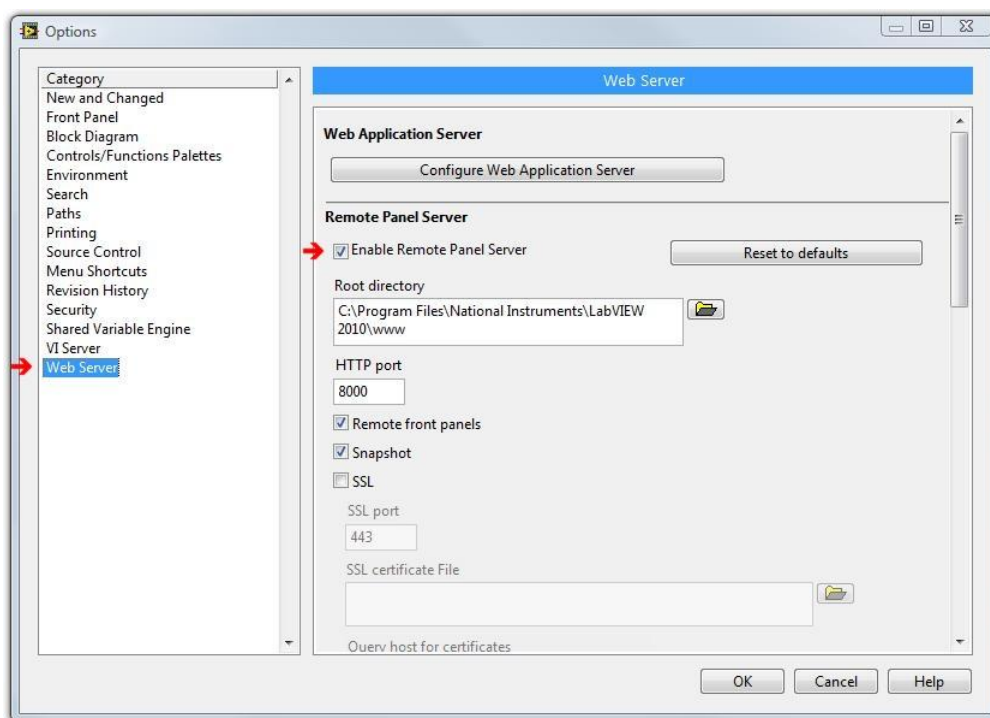
9.2 Realizace

Propojení je realizováno třemi částmi.

- nastavením parametrů pro povolení přístupu
- zapnutí WEB SERVERU
- připojení se z PC-2

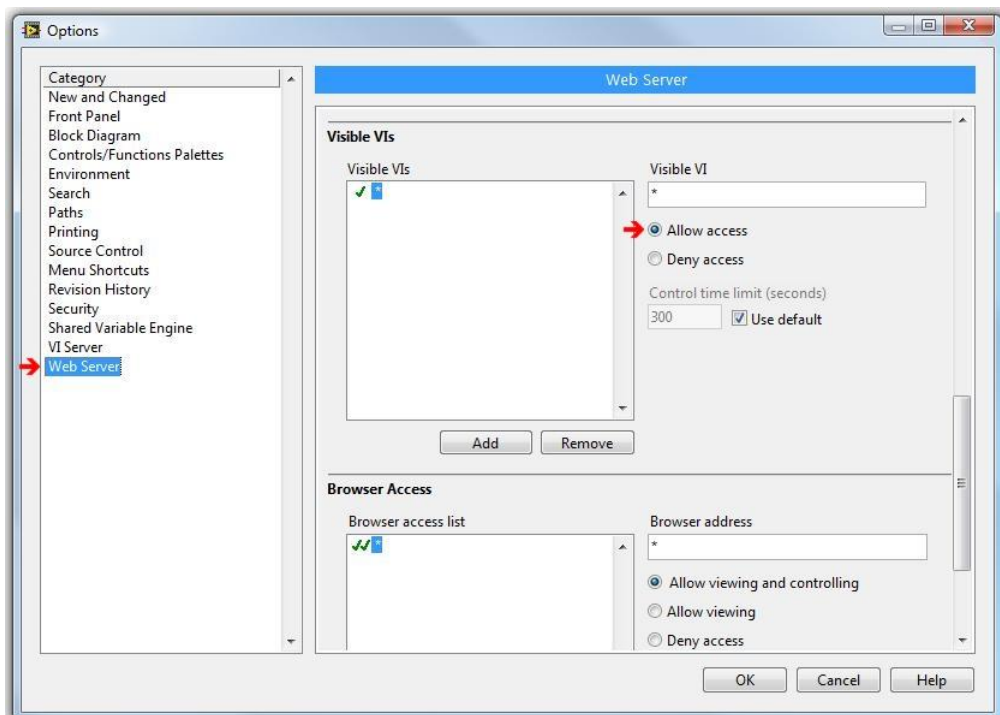
9.2.1 Umožnění přístupu z PC-1

Tato část je pouze jednorázová a po úvodním nastavení se uloží. Pomocí záložky *Tools>>Options* se otevře okno se seznamem položek v levé části (záleží na verzi LabView). Ukázka je na Obr. 53. V položce *Web Server* je nutné *Remote Panel* vůbec povolit. Učiní se tak po zaškrtnutí políčka *Enable Remote Panel Server*.



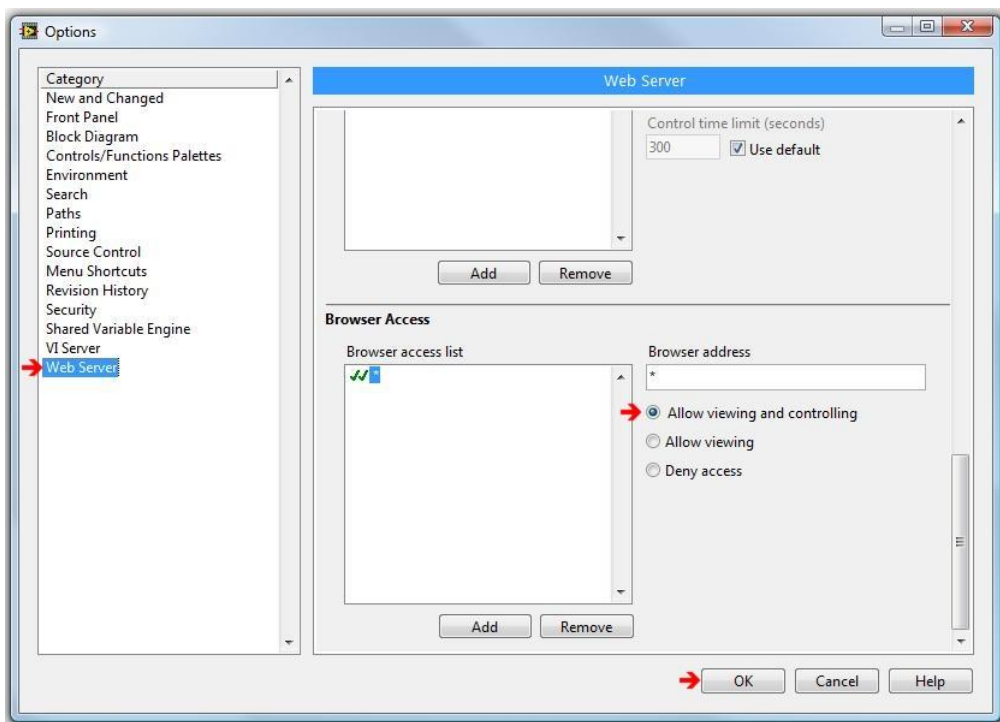
Obr. 53: Enable Remote Panel Server

Následuje krok *Visible VI's*. Zde je možné nastavit, které VI bude možné na dálku ovládat. Stačí kliknout na *Allow access* a nepřidávat žádné konkrétní VI (viz Obr. 54).



Obr. 54: Visible VI's

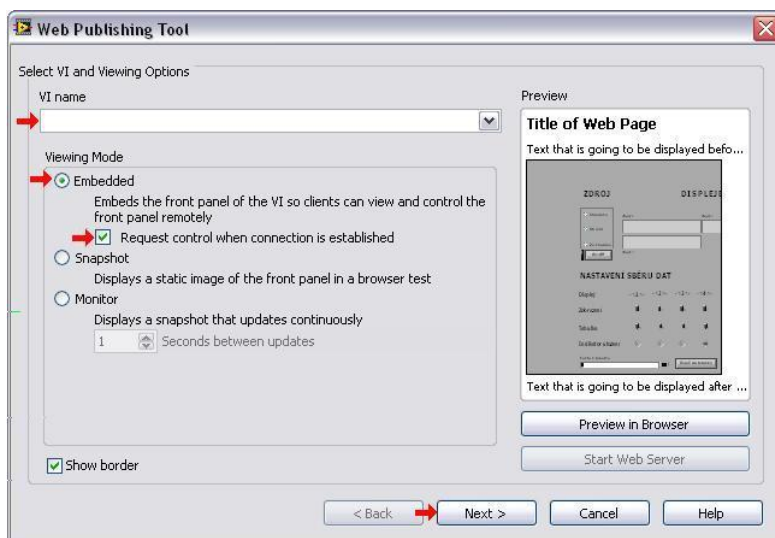
Třetím a posledním krokem je nastavení *Browser Access*. V tomto poli je možné povolit či zakázat IP adresy konkrétních pracovišť. Ty potom budou mít umožněný nebo zakázaný přístup. V tuto chvíli je nutné pouze tuto službu povolit pomocí *Allow Viewing and controlling*, neboli povolení prohlížení i řízení (viz Obr. 55).



Obr. 55: Browser Access

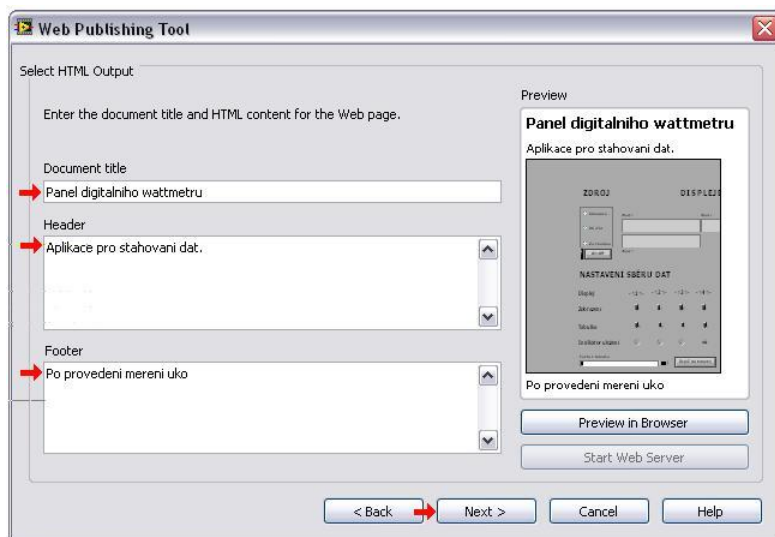
9.2.2 Zapnutí Web Serveru

Nastavení potřebných parametrů je k dispozici v záložce *Tools >> Web Publishing Tool*. V kolonce *VI name* je nutné vyplnit název programu. Vybrat *Embedded* a následně zaškrtnout položku *Request control when connection is established* (viz Obr. 56).



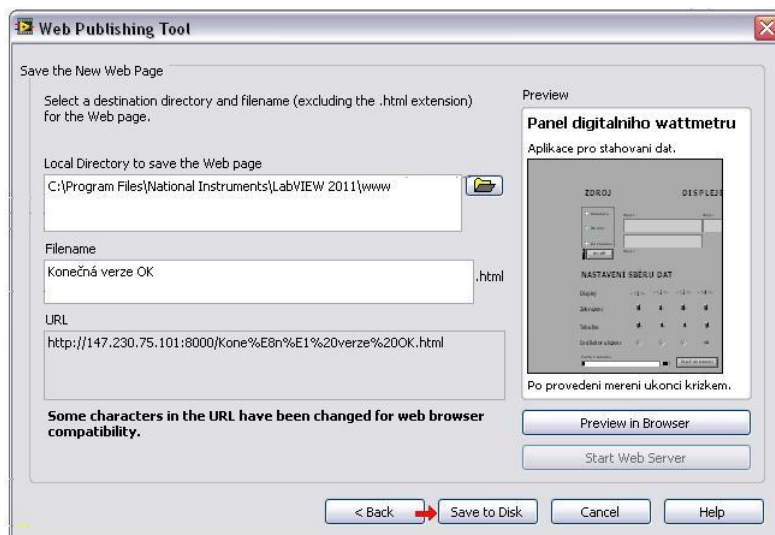
Obr. 56: Web Server VI name

V dalším okně se naskytne možnost připsat titulky (viz Obr. 57), které budou lemovat Front Panel v připojení webovým prohlížečem.



Obr. 57: Web Server titulky

Posledním krokem je uložení na disk (viz Obr. 58). Následně vyskočí okno s odkazem na vytvořenou stránku (viz Obr. 59).



Obr. 58: Web Server Save to Disk

Zešedivění tlačítka *Start Web Server* na Obr. 58 naznačuje, že tato operace již probíhá.



Obr. 59: Web Server OK

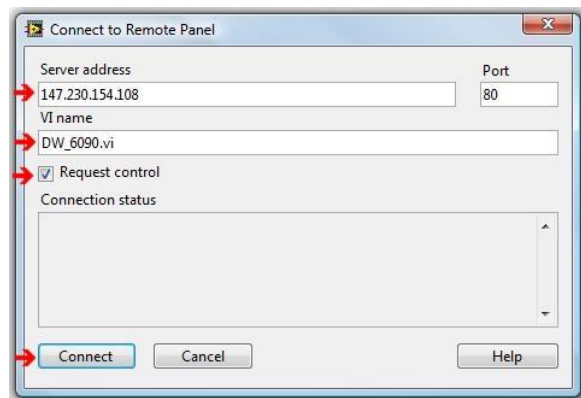
Kliknutím na *OK* se všechna okna uzavřou.

9.2.3 Navázání spojení z PC-2

Jak bylo řečeno na začátku této kapitoly, připojení je možné dvěma způsoby. Z hlediska laboratorní úlohy poskytují oba způsoby totéž.

- *Connect to Remote Panel*

Prakticky v jakémkoli otevřeném okně LabView je možné v záložce *Operate* najít položku *Connect to Remote Panel*. V následně otevřeném okně, po vyplnění IP adresy pro PC-1 v kolonce *Server address* a názvu aplikace *VI name*, je možné se připojit.



Obr. 60: Connect to Remote Panel

- *Webový prohlížeč*

Tento způsob připojení je o poznání jednodušší. Jediným úkolem uživatele je mít nainstalovaný Run-Time engine ve shodné verzi s LabView v PC-1. Poslední nutností je otevření odkazu ve webovém prohlížeči.

10 Úprava laboratorní úlohy

Prvním rozdílem od staré verze laboratorní úlohy je vzdálený přístup. Student má možnost se z domova připojit k zapojené úloze a odečíst potřebná data. Na základě této myšlenky bylo vytvořeno zadání uvedené v příloze B – Nová laboratorní úloha. Druhým zásadním rozdílem je, že je nutné opakované měření, tím dochází i ke změnám ve výpočtu nejistot.

10.1 Provedení vzorového měření

Obvod byl zapojen podle schématu uvedeného v příloze B – Nová laboratorní úloha. Pomocí regulovatelného transformátoru bylo nastaveno napětí $U = 200V$. Odečítání hodnot poté probíhalo ze vzdáleného pracoviště.

- **Použité přístroje**

Digitální wattmetr	DW – 6090	56 444 / 3
Regulovatelné trafo	STATRON	56 444 / 2
Komutátorový motor	J42RT357	126 – 056

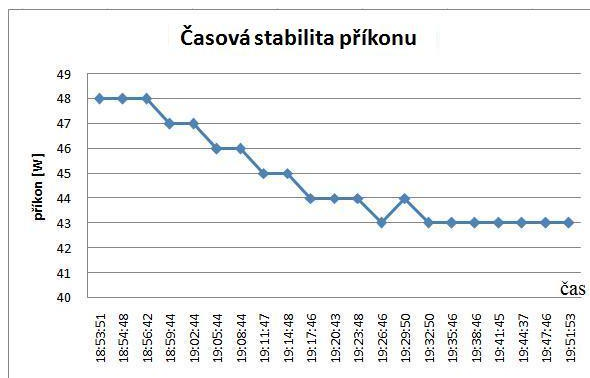
10.2 Naměřené hodnoty

Měření probíhalo přibližně jednu hodinu. Hodnoty byly zaznamenávány pomocí vzdáleného přístupu po cca třech minutách a jsou uvedeny na Obr. 61.

čas	Cos φ [-]	Příkon [W]	U [V]	I [A]
18:53:51	0,89	48	200,9	0,27
18:54:48	0,88	48	202	0,27
18:56:42	0,88	48	202	0,27
18:59:44	0,86	47	201,4	0,27
19:02:44	0,86	47	202,1	0,27
19:05:44	0,86	46	200,5	0,26
19:08:44	0,86	46	201,1	0,26
19:11:47	0,86	45	201,4	0,26
19:14:48	0,86	45	201,4	0,26
19:17:46	0,88	44	200,6	0,26
19:20:43	0,88	44	200,7	0,25
19:23:48	0,88	44	200,2	0,25
19:26:46	0,86	43	199,6	0,25
19:29:50	0,88	44	200,4	0,25
19:32:50	0,87	43	198,8	0,25
19:35:46	0,86	43	199,9	0,25
19:38:46	0,86	43	199,2	0,25
19:41:45	0,9	43	199,4	0,24
19:44:37	0,9	43	199,7	0,24
19:47:46	0,9	43	199,7	0,24
19:51:53	0,9	43	199,5	0,24

Obr. 61: Tabulka hodnot

Jak ukazuje Obr. 62, po několika minutách začal příkon klesat. První známky ustálení byly zřejmé až po 40 minutách.



Obr. 62: Časová stabilita příkonu

Účinník zprvu klesal úměrně ke klesajícímu příkonu. Později zakolísal a v závěru se opět zvýšil (viz Obr. 63).



Obr. 63: Časová stabilita účinníku

11 Shrnutí

Prvním bodem zadání bylo určeno provést rešerši. Druhý a třetí bod dával za úkol vytvořit program spolu s umožněním vzdáleného přístupu. Čtvrtým a posledním bodem zadání bylo, upravení laboratorní úlohy, která by využívala vytvořený program.

Rešerše obsahuje krátké teoretické zopakování o tom co vlastně výkon je a jakými metodami jsme schopni ho měřit. Dále je shrnuto, jakými přístroji se výkon měří. Následně je pojednáno, proč měříme výkon v praxi. Návrh připojení k přístroji byl omezený možnostmi wattmetru. Realizace byla možná pouze po sériové lince RS-232 a jako programovací prostředí bylo vybráno LabView. Výběr právě tohoto grafického programovacího prostředí se ukázal ve třetím bodě zadání jako velice prozíravý. Konfigurace sériové linky se zjednodušila do několika kroků. Většinu programu tak tvoří zpracování textového řetězce, zobrazování a zaznamenávání hodnot do tabulky. Další funkcí programu je ovladatelnost přes internet. Laboratorní úloha byla přepracována tak, aby využívala vzdáleného přístupu. Budou ji využívat studenti kombinovaného studia ke změření časové stability příkonu bez účasti ve škole. I když program vznikl právě pro upravenou úlohu, byl pojat spíše univerzálně a nebude uživatele omezovat i v měření jiných veličin.

12 Závěr

Ačkoli bylo dosaženo splnění všech bodů zadání, je možné podniknout několik kroků k určitým zlepšením. Program, který umožňuje navázání spojení, zobrazování a ukládání do tabulky, by bylo možné vyladit a dosáhnout tak rychlejšího běhu. Jedním z aspektů zpomalení je složité řešení ukládání a zobrazování hodnot. K dalšímu zpomalení dochází pouze u navázání spojení s wattmetrem. Wattmetr posílá data příliš rychle, aby bylo možné je zálohovat. V takovém případě dochází k vytváření souboru o příliš velkých rozměrech. Bylo nastaveno ukládání hodnot se zpožděním jedné vteřiny. Problém je tak vyřešen, ale zároveň se také zpomaluje i aktualizace hodnot. Jelikož se laboratorní úloha zabývá časovou stabilitou, nijak tento fakt zpomaleného čítání hodnot měření neznehodnocuje. Dochází pouze ke snížení univerzálnosti programu. Další úpravy jsou možné pouze až po zaběhnutí laboratorní úlohy. Je možné části programu ořezat a nastavit defaultně prostory pro ukládání hodnot či pevného nastavení výběru portu.

Použitá literatura

BANNERT, Petr. Měření spotřeby elektrické energie. In: *Měření spotřeby elektrické energie* [online]. Varnsdorf, 2008 [cit. 2012-04-30]. Dostupné z: http://www.vosvdf.cz/cmsb/userdata/487/FVS_014_cviceni/014_mereni_prikonu.pdf

ČMEJLA, Roman - HAVLÍČEK, Václav - ZEMÁNEK, Ivan. Základy teorie obvodů 1 : cvičení. Vyd. 2. V Praze : Nakladatelství ČVUT, 2007 dotisk. 134 s. ISBN 978-80-01-02993-0 (dotisk : brož.).

Elektrický výkon. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 24. 4. 2012 [cit. 2012-04-30]. Dostupné z: http://cs.wikipedia.org/wiki/Elektrick%C3%BD_v%C3%BDkon

Elektrotechnická měření. 1. vyd. Praha : BEN - technická literatura, 2002. 255 s. ISBN 80-7300-022-9 (brož.).

HAASZ, Vladimír - SEDLÁČEK, Miloš. Elektrická měření : přístroje a metody. Vyd. 2. Praha : Vydavatelství ČVUT, 2005 dotisk. 337 s. ISBN 80-01-02731-7 (brož.).

HORÁK, Jaroslav. *Hardware: učebnice pro pokročilé*. Brno: Computer Press, a.s., 2007. 360 s. ISBN 978-80-251-1741-5.

KAINKA, Burkhard. *Využití rozhraní PC: Měření, řízení a regulace pomocí standardních portů PC*. Ostrava : HEL, 1996. 133 s. ISBN 80-902059-3-3.

Leistungsanalysator DW 6090 D: Bedienungsanleitung. *Leistungsanalysator DW 6090 D* [online]. 2000 [cit. 2012-04-30]. Dostupné z: https://www.distrelec.com/ishop/Datasheets/iiDW6090_manual_d-f-i.pdf

PLÍVA, Zdeněk a Jindra DRÁBKOVÁ. *METODIKA ZPRACOVÁNÍ DIPLOMOVÝCH, BAKALÁŘSKÝCH A VĚDECKÝCH PRACÍ NA FM TUL* [online]. Liberec, 2009 [cit. 2012-05-07]. Dostupné z: http://www.fm.tul.cz/files/jak_psat_DP.pdf. Skripta. Technická univerzita v Liberci.

POWER ANALYZER: Model : DW-6090. *POWER ANALYZER: Model : DW-6090* [online]. 2000 [cit. 2012-04-30]. Dostupné z: https://www.distrelec.com/ishop/Datasheets/ihDW6090_data_e.pdf

Remote Panels in LabVIEW: Distributed Application Development. In: *National Instruments* [online]. 2010 [cit. 2012-04-30]. Dostupné z: <http://zone.ni.com/devzone/cda/tut/p/id/4791>

ŠNOREK, Miroslav. *Standardní rozhraní PC*. Praha: Grada a.s., 1992. 246 s. ISBN 80-85424-80-0.

VLACH, Jaroslav; HAVLÍČEK, Josef; VLACH, Martin. *Začínáme s LabVIEW*. Praha : BEN, 2008. 247 s. ISBN 978-80-7300-245-9.

VLACH, Jaroslav. *Počítačová rozhraní : Přenos dat a řídicí systémy*. Praha : BEN, 2000. 175 s. ISBN 80-7300-010-5.

Obsah CD

- Text bakalářské práce ve formátu docx a pdf
- Použité obrázky
- Krátké video s náhledem běhu programu
- Program
- Změřené hodnoty a grafy ve formátu xls
- Přílohy ve formátu pdf

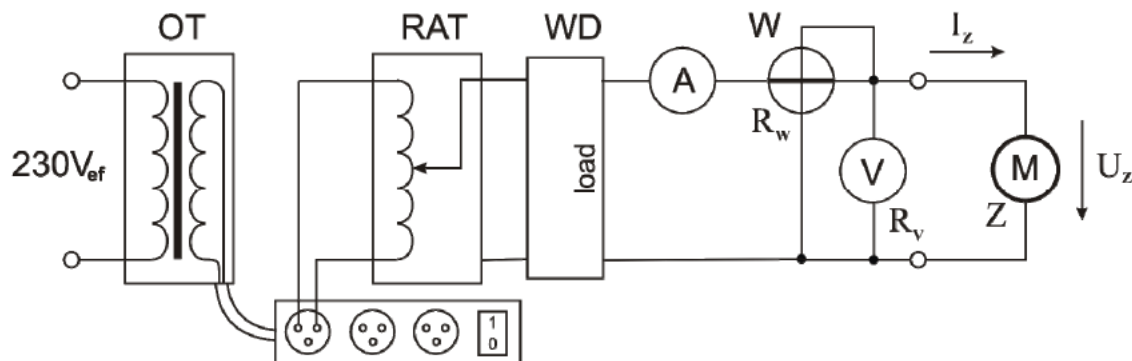
Příloha A – původní laboratorní úloha

Příkon motorku

Stanovení zdánlivého, činného a jalového příkonu a účinníku motorku pomocí wattmetrů, voltmetru a ampérmetru. Eliminace chyby metody způsobené odporem napěťové cívky wattmetru a voltmetru. Stanovení nejistoty měření.

Postup při měření:

1. Vypněte přívod 230V, zapojte přístroje dle schématu a vyčkejte kontroly vyučujícího.



M	...	komutátorový motorek univerzální
A	...	ampérmetr (Agilent U1252, rozsah 10A)
W	...	wattmetr - rozsah $U \geq 240V$; $I \geq 0,5A$
WD	...	digitální wattmetr
V	...	analogový voltmetr
OT	...	oddělovací transformátor
RAT	...	regulační autotransformátor

Koeficienty:

$\Theta = \sqrt{3}$	pro číslicové měřiče,
$\Theta = \sqrt{6}$	pro anal. měřiče (nejistota zařízení),
$\Theta = \sqrt{3}$	pro anal. měřiče (nejistota odečtu),
$k_u = 2$	koef. rozšíření pro všechny měřiče.

2. Zapněte napájení a pomocí regulačního autotransformátoru nastavte napětí U_z cca 200V (měřte analogovým voltmetrem). Na ampérmetru odečtěte celkový proud procházející obvodem, na digitálním wattmetru činný výkon a na analogovém wattmetru velikost celkového činného výkonu.
3. Vypněte napájení, rozpojte obvod a odečtěte (nebo změřte) odpor napěťové cívky wattmetru.
4. S ohledem na systematickou chybu způsobenou vnitřním odporem voltmetru R_v a odporem napěťové cívky wattmetru R_w vypočtěte skutečnou velikost zdánlivého, činného a jalového příkonu na zátěži Z (motorku), velikost proudu I_z procházejícího zátěží a hodnotu účinníku zátěže $\cos \varphi$. Pro výpočty použijte naměřené hodnoty analogovými měřiči a digitálním měřičem proudu. Digitální wattmetr použijte pouze pro srovnání hodnot naměřených ostatními měřiči.

5. Určete rozšířené nejistoty měření u vypočtených veličin (všechny typy výkonu a I_z). Při výpočtu považujte hodnoty R_v a R_w za přesné (s nulovou nejistotou).
6. V závěru zhodnoťte výsledky měření a zdůvodněte rozdílné výsledky jednotlivých měření.

Postup pro výpočet jednotlivých sledovaných veličin:

- 1) Naměřené hodnoty U'_z , I' , P' (analogový wattmetr), P_D' (digitální wattmetr)
- 2) Odpor R_v a R_w odečtete z analogových měřičů a vypočtete odpor R_p (na paralelní kombinaci wattmetr - voltmetr), nejistoty považujte za nulové:

$$R_p = \frac{R_v \cdot R_w}{R_v + R_w}$$

- 3) Proud jdoucí analogovými měřiči a jeho nejistota:

$$I_p = \frac{U_z}{R_p} \quad U(I_p) = \sqrt{\left(\frac{\partial I_p}{\partial U_z} U(U_z)\right)^2 + \left(\frac{\partial I_p}{\partial R_p} U(R_p)\right)^2}$$

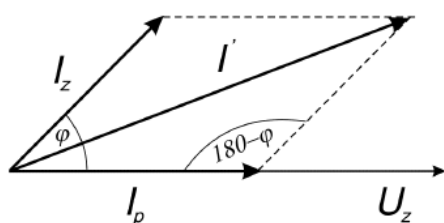
- 4) Ztrátový výkon analogových měřičů a jeho nejistota:

$$P_p = \frac{U_z^2}{R_p} \quad U(P_p) = \sqrt{\left(\frac{\partial P_p}{\partial U_z} U(U_z)\right)^2 + \left(\frac{\partial P_p}{\partial R_p} U(R_p)\right)^2}$$

Skutečný činný výkon motorku (zátěže) a jeho nejistota:

$$P_z = P' - P_p \quad U(P_z) = \sqrt{\left(\frac{\partial P_z}{\partial P'} U(P')\right)^2 + \left(\frac{\partial P_z}{\partial P_p} U(P_p)\right)^2}$$

- 5) Pro proud tekoucí motorkem platí kosinová věta:



$$I'^2 = I_p^2 + I_z^2 - 2I_p I_z \cos(180 - \varphi) = I_p^2 + I_z^2 + 2I_p I_z \cos \varphi$$

$$\left. \begin{aligned} I_p &= \frac{U_z}{R_p} \\ P_z &= U_z I_z \cos \varphi \end{aligned} \right\} \downarrow \frac{2P_z}{R_p}$$

$$I_z = \frac{P_z}{U_z \cos \varphi}$$

Pro proud I_z a jeho nejistotu pak platí:

$$I_z = \sqrt{I'^2 - I_p^2 - \frac{2P_z}{R_p}} \quad U(I_z) = \sqrt{\left(\frac{\partial I_z}{\partial I'} U(I')\right)^2 + \left(\frac{\partial I_z}{\partial I_p} U(I_p)\right)^2 + \left(\frac{\partial I_z}{\partial P_z} U(P_z)\right)^2}$$

- 6) Zdánlivý výkon motorku (příkon) a jeho nejistota:

$$S_z = U_z I_z \quad U(S_z) = \sqrt{\left(\frac{\partial S_z}{\partial U_z} U(U_z)\right)^2 + \left(\frac{\partial S_z}{\partial I_z} U(I_z)\right)^2}$$

- 7) Jalový výkon a jeho nejistota:

$$Q_z = \sqrt{S_z^2 - P_z^2} \quad U(Q_z) = \sqrt{\left(\frac{\partial Q_z}{\partial S_z} U(S_z)\right)^2 + \left(\frac{\partial Q_z}{\partial P_z} U(P_z)\right)^2}$$

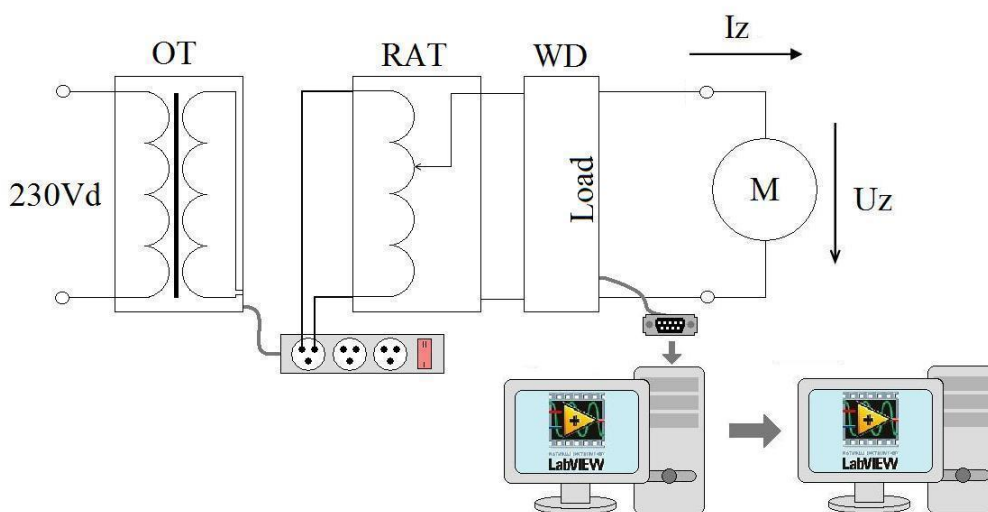
Příloha B – Nová laboratorní úloha

Příkon motorku

Stanovení časové stability příkonu a účinníku asynchronního motorku. Zaznamenávání hodnot z digitálního wattmetru DW6090 pomocí vzdáleného přístupu. Zhotovení grafů a stanovení nejistot měření.

Postup při měření:

1. Seznamte se zapojením na Obr. 1.



Obr. 1: Schéma zapojení

M ... komutátorový motorek univerzální

OT ... oddělovací transformátor

WD ... digitální wattmetr

RAT ... regulovatelný transformátor

2. Obdržený odkaz na přednášce otevřete ve webovém prohlížeči.

3. Spustíte program černou šipkou v levém horním rohu. Pod hlavičkou *ZDROJ* zaškrtněte položku *RS-232* a klikněte na tlačítko *On/Off*. Při zapnutí se rozsvítí červená dioda. V poli *SPRÁVA* vyberte v roletovém menu komunikační port. Zadejte cestu k zálohování změřených dat. Klikněte na Start.

4. Vyberte cestu k tabulce v levém dolním rohu. Tlačítkem *ZAPÍŠ DO TABULKY* zaznamenejte přibližně 20 hodnot v pravidelných intervalech v časovém rozmezí 35 až 50 minut.

5. Ze změřených hodnot zhotovte grafy časové stability příkonu a účinníku.
6. Určete nejistoty, ke kterým došlo při opakovaném měření všech měřených veličin.
7. V závěru zhodnoťte výsledky měření. Odhadněte, proč dochází k nestabilitě příkonu motorku a ke kolísání účinníku.

Příloha C – Katalogový list wattmetru DW6090

GENERAL SPECIFICATIONS				
Display	* 93 mm x 52 mm large LCD (Liquid Crystal Display) display panel.. * Multi-display unit, show Volt, Ampere, Watt, Power factor at same time.	Data Output	RS232 serial interface.	
		Sampling Time	W, A, V - 1.5 sec. DCV, DCA - 1 sec.	
		Operating Temp.	0to 50°C (32 to 122 °F).	
		Operating Humidity	Less than 80 % R.H..	
Measurement	WATT, VA, Whr,, Power factor ACV, ACA, DCV, DCA, Hz, ohm.	Power Supply	DC 9V, 1.5 V AA (UM-3) battery x 6 PCs.	
		Power Consumption	Approx. DC 30 mA	
Zero Adjustment	Watt : External adjustment.	Dimension	280 x 210 x 90 mm (11.0 x 8.3 x 3.5 inch).	
	DCV, ACV, DCA, ACA: Automatic adjustment.	Weight	Approx. 1,100 g (2.4 LB).	
Polarity	Automatic switching, "-" indicates reverse polarity.	Standard Accessories	Test lead(red & black).....1 pair.	
Current input mode	Direct input, inductive clamp probe or CT.		Instruction Manual.....1 PC.	
Over input Indication	Indication of " - - - - " or " - - - - ".			

ELECTRICAL SPECIFICATIONS (23± 5 °C)				
Watt (AC, true power), current mode from direct input.				
Range	Resolution	Accuracy	Remark	
1 Watt to 9,999 Watt	1 Watt	± (1.5 % + 1 d)	* Auto range.	
10 K Watt to 99.99 K Watt	10 Watt		* Frequency Characteristic: 45 Hz-65Hz.	
			* Accuracy is specified under input signal frequency of 50/60 Hz.	
Watt (AC, true power), current from inductive probe or CT.				
Range	Resolution	Accuracy	Remark	
1 Watt to 9,999 Watt	1 Watt	± (1.5 % + 1 d)	* Auto range.	
10 KW to 99.99 KW	10 Watt		* Frequency Characteristic: 45 Hz-65Hz.	
100.0 KW to 999.9 KW	0.1 kW		* Accuracy is specified under input signal frequency of 50/60 Hz.	
1000 KW to 1999 KW	1 kW			
VA (AC), current mode from direct input				
Range	Resolution	Accuracy	Remark	
0.01 VA to 99.99 VA	0.01 VA	± (1.5 % + 1 d)	* Auto range.	
100.0 VA to 999.9 VA	0.1 VA		* Frequency Characteristic: 45 Hz-65Hz.	
1000 VA to 9999 VA	1 VA		* Accuracy is specified under input signal frequency of 50/60 Hz.	
10.00 K VA to 39.99 kVA	10 VA			
POWER FACTOR, current mode from direct input only				
Range	Resolution	Accuracy		
0.01 to 1.00	0.01	± (1.5 % + 2 d)		
AC VOLTAGE, DC VOLTAGE				
Range	Resolution	Accuracy	Input Impedance	Overload circuit Protection
0.1 V to 299.9 V	0.1 V	± (1 % + 1 d)	10 Mega ohm	AC 700 V
300 V to 600 V	1 V			DC 1000 V
Remark : Auto range.				
AC CURRENT, DC CURRENT, current mode from direct input				
Range	Resolution	Accuracy	Voltage drop (in case of full scale)	
0.01 A to 20.00 A	10 mA	± (1 % + 1 d)	2 V AC	
* ACA, ACV frequency characteristic : 45 Hz to 65 Hz, specification be tested on sine wave of 50/60 Hz.				
* Converter response : Average responding, calibrated to display RMS value of sine wave.				
AC CURRENT DC CURRENT, current mode from inductive probe				
Range	Resolution	Accuracy	Remark	
1 A to 1000 A	1 A	± (1 % + 1 d)	ACA : 1 AC mV per 1 AC ampere. DCA : 1 DC mV per 1 DC ampere.	
AC CURRENT, current mode from CT (current transformer)				
Range	Resolution	Accuracy	Remark	
CT 100/5A, 0.1 - 100.0 A	0.1 A	Depend the CT its own accuracy.		
CT 1000/5A, 1 - 1000 A	1 A			
Watt Hour, current from direct input				
Range	Resolution	Accuracy	Remark	
0.001 Whr to 9.999 Whr	0.001 Whr	± (1 % + 1 d)	* Auto range.	
10.00 Whr to 99.99 Whr	0.01 Whr		* Frequency Characteristic: 45 Hz-65Hz.	
100.0 Whr to 999.9 Whr	0.1 Whr		* Accuracy is specified under input signal frequency of 50/60 Hz.	
1000 Whr to 9999 Whr	1 Whr			
10 K Whr to 99.99 K Whr	10 Whr			
100 K Whr to 999.9 K Whr	100 Whr			
1000 K Whr to 9999 K Whr	1 K Whr			
OHMS				
Range	Resolution	Accuracy	Overload Circuit Protection	
1 ohm to 9,999 ohm	1 ohm	± (1 % + 1 d)	AC/DC 400V	
10 K ohm to 19,99 K ohm	10 ohm			
Hz				
Range	Resolution	Accuracy	Overload Circuit Protection	
10.0 Hz to 99.9 Hz.	0.1 Hz	± (1 % + 1 d)		
100 Hz to 999 Hz.	1 Hz			

Obr. 1: Datasheet [POWER ANALYZER, 2000]